

MSX EXTRA
PVP 175 ptas. incluido IVA y sobretasa aérea Canarias
LA PRIMERA REVISTA DE MSX DE ESPAÑA
N.º 22 - Septiembre 1986 - PVP 175 ptas. (incluido IVA)
TRUCOS DEL PROGRAMADOR
Vidas infinitas para tus juegos
DEL HARD AL SOFT
Rotación y desplazamientos
MSX
Banco de pruebas:
MITSUBISHI ML-G3
Un nuevo MSX-2 para España
CALL III
Lo que ocultan los algoritmos
PROGRAMAS
Ogros
Misión imposible
Crazy Laberint
Calculadora
TODOS CON TEST DE LISTADO

SONY CONVOCA EL 2º GRAN CONCURSO DE PROGRAMAS MSX.

1er PREMIO: 1.000.000 Ptas.

Se ha abierto ya la convocatoria del 2º Gran Concurso de Programas MSX. Hay dos categorías de participación: Una, para Centros Docentes; otra para particulares y público en general.

Temario

En la categoría de "Centros Docentes" se aceptarán todos los programas cuyo tema sea pedagógico pero que, por supuesto, no sean la mera copia de un libro o de un programa ya existente. Lo que se pretende es estimular la creatividad.
En la segunda categoría, que denominamos "General", los programas que participen deberán corresponder a uno de los cuatro temas siguientes:

- ▶ Simulación en el ámbito de las Ciencias (Física, Química, Biología, Ecología, etc.). Se trata de crear un programa que simule un caso real o imaginario.
- ▶ Música (creación, interpretación, generador de sonidos y ritmos, etc.).
- ▶ Juegos de aventuras.
- ▶ Gráficos y Diseños (se valorará la posibilidad de impresión en Plotter).

Premios

Los premios se repartirán también según las categorías:

Categoría Centros Docentes.

- Un único premio de un millón de pesetas a repartir entre el Centro Docente y el autor del programa. 500.000,– Ptas. para cada uno.

Categoría General.

- Un premio de 500.000,– Ptas. para el que quede clasificado en primer lugar.
- Dos premios de 300.000,– Ptas. para los que queden clasificados en segundo lugar.
- Tres premios de 100.000,– Ptas. para los que queden clasificados en tercer lugar.

Todos los premios serán en material SONY.

Requisitos

- Los programas presentados por los Centros Docentes deberán tener un máximo de 28 K.RAM.
- Los programas presentados por particulares para la Categoría General deberán tener un máximo de 12 K.RAM.
- SONY tendrá la propiedad de los programas premiados.
- SONY tendrá los derechos de compra sobre el resto de los programas presentados.
- Los programas que concursen deberán ser presentados grabados en cinta de audio SONY o diskette SONY OM-D3440, entregándose dos copias. Asímismo se deberá adjuntar un listado del programa, instrucciones de funcionamiento y una síntesis del contenido del programa.
- Con cada programa se entregará un sobre cerrado conteniendo los datos del autor o autores, y en el exterior figurará el título correspondiente.
- Todos los concursantes, independientemente de su clasificación final, serán obsequiados con un producto SONY.

Fecha de entrega de los programas

La fecha límite para la recepción de los programas es el 30 de Enero de 1.987. Debiendo ser entregados a SONY ESPAÑA, S.A., Departamento de Ordenadores MSX. Sabino de Arana, 42-44, 08028 - Barcelona; TEL. (93) 330.65.51.

Fallo del concurso y entrega de premios

Entre todos los programas recibidos, el jurado elegirá los que, a su juicio, contengan un mayor nivel de innovación y creatividad.

El fallo será público el 1 de Abril de 1.987 y publicado en la prensa nacional.
Para mayor información o consulta, diríjase a cualquiera de las Delegaciones SONY.

Los Sres. Juan Roig Ferrán de Constantí (Tarragona) Jesús Asín Gascón de Salamanca, y Enrique Riera Quiles de Valencia fueron ganadores del Primer Concurso de Programas MSX. Sus programas han sido publicados por SONY y actualmente están siendo comercializados. Con los ganadores de este año se hará lo mismo. Tú puedes ser uno de ellos.

MSX

ORDENADORES
HIT BIT

SONY®

DELEGACIONES SONY ESPAÑA, S.A.

BARCELONA
Sabino de Arana, 42-44
Tel. (93) 330 65 51
08028 BARCELONA

MADRID
Julian Romea, 8
Tel. (91) 253 08 00
28003 MADRID

BILBAO
Pintor Lecuona, 1
Tel. (94) 444 42 00
48012 BILBAO

SEVILLA
Niebla, 8
Tel. (954) 27 47 07
41011 SEVILLA

VALENCIA
Salvador Ferrandis Luna, 6
Tel. (96) 325 35 06
46018 VALENCIA

LA CORUÑA
Avda. Ejército, 23
Tel. (981) 29 98 55
15006 LA CORUÑA

Editorial

HOLA SONIMAG

Un año más saludamos a SONIMAG. No es un saludo convencional, sino la expresión de sincera alegría por contar con uno de los salones dedicados a la imagen, el sonido y la electrónica más importantes de Europa. En esta su 24.ª edición SONIMAG camina firmemente con el apoyo de decenas de grandes empresas nacionales e internacionales, que tienen en él un inestimable escaparate para sus productos.
Para ello sus miembros directores han desarrollado una política basada en la seriedad y en la actualidad ya nadie es capaz de discutir el prestigio del Salón Internacional de la Imagen, el Sonido y la Electrónica, que se celebra anualmente en el recinto de la Feria de Barcelona.
No hay acontecimiento tecnológico producido en el mundo desde su creación que no haya sido reflejado en SONIMAG. En este sentido era lógico que el segmento doméstico de la informática estuviese presente de una manera cada vez más notable.
Manhattan Transfer, S.A., consciente del momento singular que vivimos, en el umbral mismo de una nueva era, estará presente por segundo año consecutivo. Pero no será una mera presencia testimonial. También el sistema en el que creemos tiene mayores posibilidades, por su configuración, de comunicar a todos los ordenadores domésticos y por ende a los hombres, vive un momento muy especial. Por ello nuestra presencia será muy activa, ya que en el marco espectacular de SONIMAG presentaremos las PRIMERAS JORNADAS DEL MSX con el inestimable apoyo de todos los sectores comprometidos con el MSX. Durante esas jornadas queremos que todos nos veamos las caras, dilucidemos dudas y expongamos los lineamientos generales de una política coherente, en los campos comercial, productivo y asistencial. Una vez más SONIMAG es la gran oportunidad. Por ello, y por todo lo que significa para el progreso español y su inserción en el Mercado Común, saludamos entrañablemente a SONIMAG 86.

MANHATTAN TRANSFER, S.A.



SUMARIO

AÑO II N.º 22 SEPTIEMBRE 1986
P.V.P. 175 ptas. (incluido IVA y sobretasa aérea Canarias)
Aparece los días 15 de cada mes.

SUPER JUEGOS EXTRA MSX ES EDITADA POR MANHATTAN TRANSFER, S.A.
Director Editorial: Antonio Tello Salvatierra.
Director Ejecutivo: Birgitta Sandberg.
Redactor Jefe: Javier Guerrero. **Redactores:** Claudia T. Helbling, Silvestre Fernández y Rubén Jiménez. **Colaboradores:** Santiago Guillén, Fco. Jesús Viceyra, Joaquín López. **Departamento de programación:** Juan C. González. **Diseño:** Félix Llanos. **Grafismo:** Juan Núñez, Carles Rubio. **Suscripciones:** Silvia Soler. **Redacción, Administración y Publicidad:** Roca i Batlle, 10-12. 08023 Barcelona. Tel. (93) 211 22 56.
Fotomecánica y Fotocomposición: Ungraf, S.A. Pujadas, 77-79. 08005 Barcelona.
Imprime: Rotedic, S.A. Ctra. de Irún, km. 12,450. Fuencarral. 28049 Madrid.
Distribuye: GMA, S.A. Eduardo Torroja, 9-11. Fuenlabrada (Madrid). Tel. (91) 690 40 01

LETRAS PARPADEANTES

Les escribo para preguntarles cómo se puede escribir con letras parpadeantes en SCREEN 1. También les animo para que hagan un apartado en la revista de código máquina.

Fernando Martín Consuegra
Majadahonda (Madrid)

En principio, el VDP de los MSX no puede representar caracteres parpadeantes. Sin embargo, existe la posibilidad de simularlo, en SCREEN 1, manipulando la zona del color. Esta zona empieza en la dirección 8192 y tiene una longitud de 32 bytes. Puesto que es posible mostrar hasta 256 caracteres diferentes, cada uno de los 32 bytes de la zona de atributos indica el color de un grupo de 8 figuras. Así, manipular el contenido de la dirección de la VRAM 8192 cambia el color de los caracteres cuyos códigos gráficos van del 0 al 7; tocar la posición 8193 hace variar los caracteres del 8 al 15; etc.

La siguiente rutina sirve para mostrar el contenido de una cadena (A$) con un efecto de FLASH. El programa empieza creando una variable falsa (B$), que contiene los códigos de las figuras 232 a 247. A continuación se copian las definiciones de cada uno de los caracteres que componen la cadena A$ en la zona de patrones que representará B$. Por último, la línea 150 se encarga de conmutar los colores de la tinta y el fondo para obtener el efecto de FLASH.

```
10 SCREEN1
20 FORX=232TO247:
B$=B$+CHR$(X):NEX
T
30 INPUTA$:GOSUB1
10:END
100 'RUTINA FLASH
110 FORX=1TOLEN(A
$):FORY=0TO7
120 VPOKE(231+X)*
8+Y,VPEEK((ASC(MI
D$(A$,X,1)))*8+Y)
130 NEXTY,X
140 PRINTLEFT$(B$
,LEN(A$))
150 VPOKE8221,VPE
EK(8221)/16+(VPEE
K(8221)AND15)*16:
VPOKE8222,VPEEK(8
221)
160 X=X+1:Z$=INKE
Y$:IFZ$<>""THENRE
TURN
170 IFX<30THEN160
ELSEX=0:GOTO150
```

Observa que para abandonar la rutina basta con pulsar una tecla. Esto, evidentemente, puede ser cambiado para adaptarlo a tus preferencias, así como la cantidad máxima de caracteres que puede manejar el programa (16, en este caso). Por otra parte, la línea 170 se encarga de controlar la frecuencia del parpadeo. Prueba a modificar el número 30 que hay tras «IF X<» y obtendrás un FLASH más o menos rápido.

En cuanto a tu sugerencia sobre el apartado del código máquina, habrás observado que ya hay varias secciones que tocan el tema habitualmente.

PROBLEMAS CON EL DISCO

Tengo un HIT BIT HB 75-B y una unidad de discos PHILIPS VY0010. Mi problema es que ningún programa en código máquina me funciona cuando tengo la unidad de disco conectada. Si no hay posibilidad de que estos programas funcionen, debemos anotar un punto muy negativo de los ordenadores MSX, ya que esto no sucede en los ordenadores AMSTRAD.

También quisiera saber cómo puedo programar el ordenador para que al conectar la unidad de discos autoejecute un programa, es decir, que no haya necesidad de teclear LOAD«....», R.

Gonzalo Satorre Miralles
(Valencia)

No comparto tu opinión en lo referente al punto negativo que mencionas, puesto que un programa que viene en cassette no tiene por qué garantizar que corra en un ordenador con disco. Personalmente trabajo con frecuencia con un AMSTRAD 6128 y puedo asegurarte que si un programa «machaca» las variables del sistema o el buffer de algo más de 1K que emplea el disco, es seguro que ocurrirá alguna desgracia. Tal es el caso de algunas cintas de su hermano pequeño (el AMSTRAD 664), que no dispone de floppy. Por otra parte, si conoces la unidad de discos de los AMSTRAD, y has trabajado con programas devoradores de Kbytes, convendrás conmigo que lo mejor que se puede decir de ella es que «cuesta poco dinero».

En fin, para solucionar el problema en los MSX y conseguir utilizar los programas desde el disco, yo suelo emplear un pequeño programa cargador, que sirve para eliminar la unidad en cuanto los bytes están en la memoria:

```
10 BLOAD"NOMBRE.B
IN"
20 Z$=INKEY$:IFZ$
=""THEN20
30 FORX=&HFD9ATO&
HFFFE:POKEX,201:N
EXT
40 CLEAR200,&HF37
F
50 DEFUSR=DIRECCI
ON:A=USR(0)
```

La línea 20 es necesaria, puesto que si no existiera se correría el riesgo de eliminar la unidad (líneas 30 y 40) mientras el motor aún está en marcha. Por tanto, espero hasta que el indicador de funcionamiento se apaga, para continuar, pulsando una tecla. De cualquier forma, hay bastantes programas comerciales que inician su ejecución en la dirección 32768, que, como sabes, es el comienzo del área del BASIC. Puesto que esto impide en parte el empleo de un cargador, modifico la línea 10, añadiendo un desplazamiento de los bytes hacia arriba, y cambio la dirección de ejecución de la línea 50, para dirigirla hacia una pequeña rutina en código máquina que hace lo siguiente:

```
LD  DE,32768
LD  HL,32768+DES-
PLAZAMIENTO
LD  BC,LONGITUD
DEL BLOQUE
LDIR
JP  DIRECCION
EJECUCION
```

Como ves, se trata simplemente de mover, con la instrucción LDIR, todos lo bytes del programa a su posición original. Quizá te parezca trabajoso, pero es algo que sólo debe hacerse una vez por programa.

En cuanto a cómo autoejecutar un programa desde el disco, puedes hacerlo si lo grabas con el nombre de AUTOEXEC.BAS.

SLOTS

Quisiera formular algunas preguntas y hacer una sugerencia:

¿Qué utilidad tiene cambiar las páginas de un slot a otro?

¿Podrían indicarme cuál es la disposición de dichas páginas en el CANON V-20?

En el programa DESEMSAMBLADOR del N.º 19-20, para acceder a la VRAM, ¿qué tengo que poner en las preguntas «dirección inicial y final»?

Me gustaría saber dónde se puede conseguir el nuevo catálogo de software MSX y su precio.

He oído que hay otros ordenadores que no son MSX que también utilizan el procesador Z80. ¿Esto quiere decir que tiene la misma disposición de memoria? ¿Cuáles son las direcciones de memoria donde se almacenan los datos de los SPRITES (forma, color, etc.)?

Me gustaría que en la nueva sección CALL dedicaran un número a cómo conseguir SPRITES multicolores y a cómo lograr plasmar en un segundo una pantalla llena de gráficos (como hacen los programas comerciales en código máquina, donde al cambiar de pantalla ésta no se va dibujando como lo hace el BASIC, sino que lo hace de pronto).

Pablo Pardo Márquez
Granollers (Barcelona)

—El Z80 sólo es capaz de direccionar 64K. Esto quiere decir que un momento determinado sólo pueden haber 64K a la vista del microprocesador. Puesto que los MSX admiten la conexión de cartuchos (de ROM y de RAM) y existen programas que necesitan más memoria que la direccionable por el Z80, se ha previsto un circuito que conmute los diferentes bancos de memoria (el chip PPI 8225 de Intel), a fin de poder superar las limitaciones del microprocesador. De esta forma, es normal que un programador emplee, simultáneamente; 32K del BIOS y del BASIC; 32K de la RAM del usuario; 16K del cartucho de control de la unidad de discos y 32 K de la RAM «semioculta», que existe en todos los aparatos de 64K, para contener datos o un programa de utilidades (un ensamblador, por ejemplo).

—Para saber los slots activos puedes leer el registro de selección de ranura con la función BASIC INP, empleando una línea como esta:

10 PRINT BIN$ [INP(&HA8)]

Los dos bites menos significativos indican el slot seleccionado de la página 0, en un número comprendido entre 0 y 3. Los siguientes dos bits indican el slot de la página 1, etc.

—El programa desensamblador que mencionas (o cualquier otro) no actúa sobre la VRAM. De cualquier forma, desensamblar la RAM de vídeo sería del todo inútil, puesto que los datos que obtendrías carecerían de sentido.

—Parece ser que todos los fabricantes relacionados con el MSX van a presentar un catálogo conjunto de software, pero en el momento de cerrar esta edición no había nada definitivo. Sin embargo, te recomiendo que consigas un ejemplar de nuestro ESPECIAL SOFTWARE, pues es seguro que encontrarás lo que buscas.

—El hecho de que varios ordenadores empleen el Z80 como CPU no implica que tengan la misma disposición de memoria. En realidad pueden tener una arquitectura completamente diferente y sus sistemas operativos pueden ser muy dispares. Lo que sí tienen en común es que responden a los mismos códigos de ensamblador.

—En cuanto a tu sugerencia, por el momento no aparecerá un apartado de la sección CALL dedicado a los SPRITES, aunque en el número de agosto de nuestra revista hermana, MSX CLUB, se inició una serie de artículos que tocan el tema en profundidad.

Por otra parte, plasmar instantáneamente un dibujo en la pantalla es muy fácil. La rutina del BIOS situada en &H41 sirve para desconectar la pantalla y ponerla del color del fondo, aunque todo lo que envíes a ella se almacenará y podrás ver el dibujo completo reactivándola con una llamada a la rutina de la posición &H44. Esto se consigue con DEFUSR = &H41:A= USR (0), para apagarla, y DEFUSR=&H44:a=usr (0), para volverla a encender. Alternativamente, puede hacerse desde el BASIC, manipulando el registro uno del VDP. Así, VDP(1) = VDP(1)AND191 la desconecta y VDP(= VDP(1) de =(1) OR64 la hace visible de nuevo.

MSX-2 A TOPE

Esta computadora de avanzada es fruto de la investigación tecnológica de MITSUBISHI durante los muchos años que lleva esta firma en el terreno de los ordenadores. El ML-G3 es un ordenador personal de 8 bits desarrollado de acuerdo al standard MSX2 que ofrece las prestaciones máximas en cuanto a computadoras de su categoría.

MSX2

El MITSUBISHI ML-G3 es el de máximas prestaciones de los dos que ha lanzado esta firma recientemente al mercado.

El equipo completo consta de dos módulos: la unidad principal, donde se alojan todas las conexiones y el teclado, independiente y extraplano –recuerda el de un IBM PC– que se conecta a la unidad principal mediante un cable extensible, lo que permite tener a esta última colocada en otro lugar (por ejemplo en el rack del equipo audio), mientras se trabaja cómodamente con el teclado.

El teclado de este equipo tiene el mismo diseño que el de los modelos ML-FX1/ML-FX2 con la salvedad de que es más ligero y delgado al no albergar la fuente de alimentación ni otros componentes electrónicos Como recordamos de sus antecesores de la primera generación, antes mencionados, la parte derecha del teclado alberga –además de a los cursores– un teclado numérico muy útil para la introducción de cantidades, y expresiones aritméticas. Todo el hardware viene completado con dos cables, uno de conexión RGB analógico y uno de RF para conectar bien a un monitor o a un aparato de televisión.

UNIDAD PRINCIPAL

El corazón de este equipo lo constituye la unidad principal, donde además de alojarse la CPU y el resto de la circuitería se hallan todos los interruptores, ranuras y conexiones además de la primera unidad de disco.

En la parte delantera de la unidad principal, y de izquierda a derecha, nos encontramos con la abertura del cartucho, el espacio previsto para el montaje de la unidad adicional de disco, y la unidad principal de disco. Esta unidad de disco ya nos resulta conocida, puesto que se trata de la misma que comercializa MITSUBISHI para los aparatos de la primera generación y como recordamos trabaja con disketes de 3'5" de simple o doble cara con una capacidad de 1Mb. Debajo de la ranura del cartucho se encuentra el interruptor de encendido con su correspondiente lámpara indicadora, y debajo de la unidad de disco el interruptor de RESET.

En la parte posterior, la unidad principal alberga la mayor parte de las conexiones. En la parte superior izquierda se encuentran todas las conexiones audio/vídeo montadas sobre una placa móvil

ESPECIFICACIONES

Item			Especificaciones
CPU			Equivalente al Z-80A (frecuencia de reloj 3,57945 MHz)
Memoria	ROM	BASIC	32 kB
		BASIC expandido	16 kB
		DISK-BASIC	16 kB
		RS-232C	8 kB
	RAM	RAM principal	64 kB
		Video RAM	128 kB
Indicaciones en pantalla	Modo de texto		32 car. x 24 líneas (16 colores entre 512) 40 car. x 24 (líneas (2 colores entre 512) 80 car. x 24 líneas (2 colores entre 512)
	Modo gráfico de alta definición		256 x 192 puntos (16 colores entre 512)
	Modo gráfico de alta definición 2		256 x 192 puntos (16 colores entre 512)
	Modo gráfico de baja definición		64 x 48 bloques (16 colores entre 512)
	Modo gráfico por mapa de bits		256 x 212 (16 colores entre 512) 512 x 212 (4 colores entre 512) 512 x 212 (16 colores entre 512) 512 x 424 (16 colores entre 512, modo de entrelazado) 256 x 212 (256 colores)
	Caracteres		Matriz de 5 x 7 puntos Letras, número y signos 94 Códigos 71 Signos gráficos 87
Teclado	Teclas		Distribución MSX española Lestas y números Letras especiales Teclas de código y edición Teclas especiales Teclas muerta Teclas de cursor

Item		Especificaciones	
Teclado	Teclas	5 teclas de función (10 funciones posibles mediante la tecla SHIFT)	
	Tipo	Separado de la unidad principal	
Generación de sonido		8 octavas y efectos sonoros Acordes de tres notas Volúmen ajustable por software	
Reloj		Indicación de año, mes, dia, hora, minutos y segundos (con pila de apoyo)	
Unidad de disco flexible	Capacidad de memoria	Sin formatear	1 MByte
		Formateado	720 kByte
	Volocidad de transmisón	250 kbits/seg.	
	Tiempo de acceso	Pista – pista	12 ms
		Promedio	331 ms (incluyendo búsqueda y estabilización)
		Tiempo de estabilización	15 ms
		Tiempo de accionamiento de otor	400 ms (600 ms máx. hasta la salida de la señal Ready)
	Densidad de registro	8717 BPP (bits por pulgada)	
	Sistema de registro	MFM	
	Densidad de pistas	135 PPP (pistas por pulgada)	
	Número de cilindros	80	
	Número de cabezas	2	
	Velocidad de rotación	300 rpm (revoluciones por minute)	
	Discos empleados	3,5 pulgadas, ambos lados (o un lado sólo), 135 PPP, tipo estándar	
	Número de unidades conectables	2 como máximo	
	Formateado	512 bytes/sector, 9 sectores/pista, 80 pistas/cabeza	



prevista para futuras expansiones. Las terminales que se encuentran sobre esta placa son las siguientes: Conexión de RGB, terminal de salida de vídeo (vídeo out), salida de audio (Audio Out) y terminal RF. A la derecha de ellos se encuentra la terminal de la interfaz RS-232 C prevista para comunicaciones, que dispone de un BASIC extendido para su uso. En la parte inferior izquierda se encuentra el interruptor de la interfaz, y a su derecha la terminal de conexión de impresora y la conexión de cassette.

En su lado izquierdo, se ubican los ports de los joysticks y la terminal de conexión del teclado.

CARACTERISTICAS DEL EQUIPO

Las especificaciones del MSX2, a las que responde este equipo totalmente son las siguientes:

La CPU es nuestro viejo conocido el mircroprocesador z80A, y en cuanto a la memoria y su distribución es la siguiente:

La ROM tiene 72Kb repartidos del siguiente modo:

32K BASIC
16K BASIC expandido
16K DISK-BASIC
8K RS 232-C

Item		Especificaciones
Interfase de comunicaciones RS-232C	Sinoronización	Sincronización choho?
	Velocidad de transmisión	50, 75, 110, 300, 600, 1200, 1800, 2000, 2400, 3600, 4800, 7200, 9600, 19200 BPS (ajustable a distintos valores para emisión y recepción)
	Bits de comienzo	1 bit
	Bits por carácter	5, 6, 7 y 8 bits
	Bit de paridad	Par, impar o ninguno
Interfase RS-232C	Bits de parada	1 – 1,5 – 2 bits
	Nivel de señal	ON: de + 5 a + 12 V; OFF: de –5 a – 12 V
	Sistema de transmisión	Dúplex completo
	LSI empleado	Interfase programable de comunicaciones: Equivalente a la I 8251 Temporizador programable de intervalos: Equivalente al I 8253
Interfase de grabador de cassettes		Velocidad de transferencia: 1200/2400 baudios
Interfase de impresora		Interfase paralela de 8 bits
Interfase de palanca de comando		Utilizable también para ratón
Terminales de conexión		Ranuras externas 1, 5 espigas Ranuras internas 2, 50 espigas Terminal de impresora 1, 14 espigas Terminales de palanca de comando 2, 9 espigas Terminal RGB snalógica de salida 1, 8 espigas Terminal de salida de RF 1, 2 espitas Terminal de salida compuesta de video 1, 2 espigas Terminal de salida de audio 1, 2 espigas

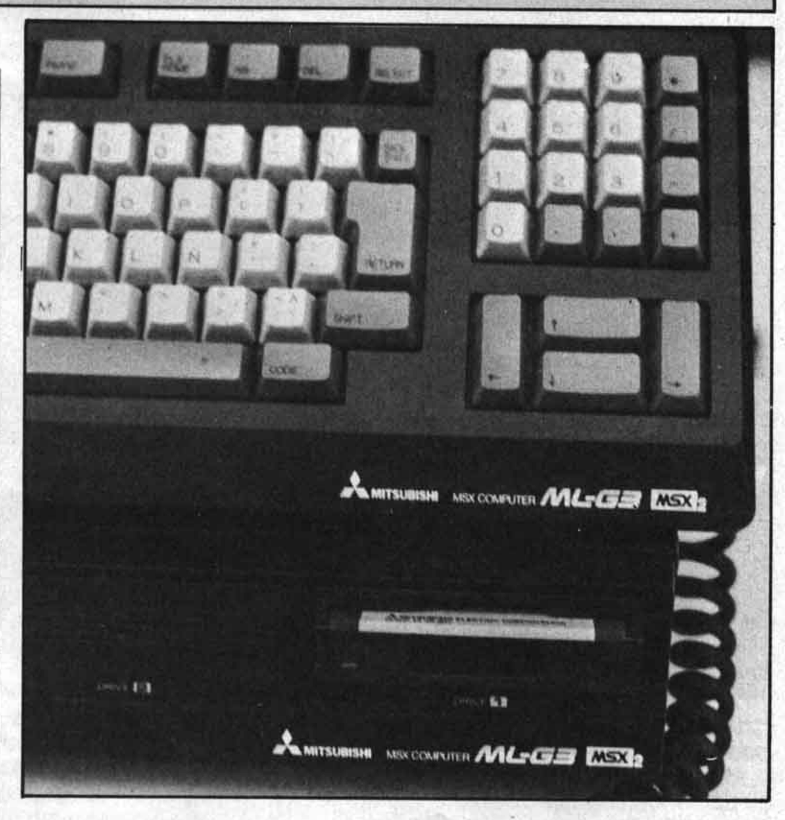


Item		Especificaciones
		Terminal de E/S de grabador 1, 8 espigas Terminal de interfase RS-232C 1, 25 espigas
Opcionales (incorporables a la unidad)		Unidad de disco flexible de 3,5 pulgadas (hasta 2 en total)
Requerimientos de energía		220V CA, 50 Hz
Consumo		31 W
Condiciones Ambientales	Temperatura	10 – 35°C
	Humedad	20 – 80% (libre de rocío)
Dimensiones	Unidad principal	360 (An) x 368 (Pr) x 98 (Al) mm
	Teclado	412 (An) x 190 (Pr) x 34 (Al) mm
Peso	Unidad principal	5,9 kg
	Teclado	1, 3 kg

MODOS DE PANTALLA

El MSX2-BASIC tiene 9 modos SCREEN, que ofrecen diferentes definiciones de pantalla y la función de visualización.

N.º de modo	Modo	Visualización en pantalla	Color	Página	Figura móvil
Ø	Texto	Máximo de 80 caracteres horizontales, 24 líneas verticales	Función de paleta de colores, 16 colores/512 colores	—	No se emplea
1		Máximo de 32 caracteres horizontales, 24 líneas verticales	Función de paleta de colores, 16 colores/512 colores	—	Se emplea
2	Gráficos, VRAM de 64 o 128K	256×192 puntos	Función de paleta de colores, 16 colores/512 colores (2 colores/8 puntos)	—	Se emplea
3		Multicolor de 64×48 puntos	Función de paleta de colores, 16 colores/512 colores	—	Se emplea
4		256×192 puntos	Función de paleta de colores, 16 colores/512 colores (2 colores/8 puntos)	—	Se emplea (figura móvil ampliada)
5		256×212 puntos	Función de paleta de colores, 16 colores/512 colores	2 páginas (VRAM de 64K), 4 páginas (VRAM de 128K)	Se emplea (figura móvil ampliada)
6		512×212 puntos	Función de paleta de colores, 4 colores/512 colores	2 páginas (VRAM de 64K), 4 páginas (VRAM de 128K)	Se emplea (figura móvil ampliada)
7	Gráficos VRAM de 128K solamente	512×212 puntos	Función de paleta de colores, 16 colores/512 colores	2 páginas	Se emplea (figura móvil ampliada)
8		256×212 puntos	256 colores	2 páginas	Se emplea (figura móvil ampliada)



La RAM es de 192K, que se reparten en 64K de RAM principal y 128 de VRAM, lo que nos da un máximo de 9 modos de pantalla (de SCREEN 0 a SCREEN 8), correspondiendo los dos primeros a modo de texto y los otros siete a gráficos. En SCREEN 0, se puede seleccionar la anchura de la línea a 40 u 80 columnas indistintamente además de todos los valores intermedios, lo que permite –entre otras cosas– trabajar con el sistema operativo CP/M.

En los 7 modos gráficos, las posibilidades son inverosímiles. Se puede seleccionar desde un modo de 64x48 puntos de definición, con la posibilidad de emplear en él hasta 16 colores de entre 512, otro de 512x212 puntos con 4 colores entre 512, o bien otro de 256x212 puntos con 256 colores. (Ver figura 1).

Además de estas, y muchas otras capacidades del equipo, que irán descubriendo los usuarios paso a paso y acerca de las cuales hablaremos largo y tendido desde estas páginas, y sobre las cuales pueden ir haciéndose una idea si observan los cuadros de especificaciones del equipo que acompañan este artículo, merece la pena resaltar la inclusión de unos completos manuales –de instrucciones generales, del lenguaje BASIC, del MSX-DISK BASIC, del BASIC expandido para comunicaciones mediante el RS-232C y del MELBRAINS NOTE. Pero, ¿qué es el MELBRAINS NOTE?



MELBRAINS NOTE

Se nos había olvidado comentar, que este equipo viene acompañado de un diskette con el sistema operativo MSX DOS, del que también hay manual, por supuesto, y que lleva de regalo algo a lo que nos viene acostumbrando MITSUBISHI. Un programa integrado de Gestión denominado MELBRAINS NOTE, que de alguna manera recuerda al MAP del ML-FX2.

Este programa contiene un potente procesador de textos y un Graphic Tool (herramienta gráfica textualmente) que va a resultar muy útil a aquellos que adquieran este equipo.

La carga de este programa se realiza desde el MSX DOS, y mediante el mismo se puede preparar un diskette para realizar una copia de seguridad de este programa, con las indicaciones que facilita Mitsubishi.

En resumen, este equipo comercializado por MABEL S.A. a un precio que colmará con creces las expectativas de aquellos usuarios que esperaban con ansiedad la aparición de equipos de la segunda generación de MSX en el mercado.

ALGORITMOS

La tercera entrega de esta sección aborda un tema de cuya claridad depende gran parte de la comprensión que se tendrá posteriormente de los mecanismos informáticos.

Algoritmo es una palabra que se oye con cierta frecuencia en el mundillo informático. Es posible que muchos de vosotros no sepáis qué se esconde realmente tras ella. Pues bien, si buscáis su significado en un pequeño diccionario encontraréis una definición parecida a ésta: «notación propia de una forma particular del cálculo». De lo anterior se deduce que el término «algoritmo» no es exclusivo de la jerga informática, en realidad es una palabra derivada del nombre de un matemático árabe llamado Al-Khuwarizmi, y es en ese campo donde se emplea desde mucho antes de la aparición de los ordenadores electrónicos.

Ciñéndonos a nuestro medio, podemos definir un algoritmo como un grupo de instrucciones concretas que sirve para resolver un problema determinado. Todo esto conlleva que realizar un programa, en cualquier lenguaje, no sea más que construir un algoritmo adecuado que resuelva el problema.

Muchos de los algoritmos que se emplean en la programación de ordenadores, y en cualquier otro contexto, fueron realizados siglos antes de la aparición del primer transistor. Gran cantidad de personas, a lo largo de la historia, ha exprimido su cerebro para solventar problemas comunes que, de otra forma, se harían casi intratables. Sin más, imagina cómo podrías hacer una simple división si no conocieras el algoritmo que todos aprendimos en el colegio, consistente en ir obteniendo dividendos parciales, que a su vez se forman añadiendo al resto la siguiente cifra del dividendo original. Quizá pienses que saber dividir no es importante para programar bien, porque cualquier artefacto electrónico lo hace con suma rapidez. Pues bien, confío en demostrarte lo contrario a lo largo de este artículo.

CONSTRUCCION DE ALGORITMOS

El problema de los algoritmos es que no existen reglas fijas para su construcción, puesto que cada uno de ellos va dirigido a un problema particular. No obstante, un programa, por complejo que sea, suele echar mano de pequeños algoritmos, ya comprobados, que facilitan enormemente la tarea del programador. En la práctica, suelen construirse algoritmos que se limitan a resolver las cuestiones planteadas, sin mirar mucho la optimización de los mismos. De esta forma, es posible que un programador haga una aplicación que funcione varias veces más lenta y derroche bastante más memoria que el mismo algoritmo completamente optimizado.

Otro factor importante a la hora de implementar algoritmos es conocer los medios de los que se dispone. Naturalmente no tendría sentido construir una rutina para obtener la raíz cuadrada de un número si se piensa trabajar en BASIC, porque el intérprete es capaz de hacerlo por sí solo.

UN EJEMPLO PRACTICO

Para profundizar en lo expuesto hasta ahora, construiremos un algoritmo que se encargará simplemente, de multiplicar dos números de 16 bits desde el código máquina. El motivo por el que he elegido un algoritmo de multiplicación es que resulta bastante didáctico y su implantación es necesaria en la práctica totalidad de los programas; no hay más que decir que la mayoría de los juegos que se comercializan no pueden permitirse perder el tiempo que el basic derrocha, y deben incorporar una rutina como la que se describe al final.

Como ya sabéis, el Z80 sólo es capaz de efectuar sumas y restas (de números comprendidos entre 0 y 65535 ó –32768 y 32767), no así multiplicaciones o divisiones. Por consiguiente, deberemos valernos de su facultad de sumar para conseguir la multiplicación.

Estoy casi seguro de que todos los recién llegados al ASSEMBLER que construyen una rutina rápida de multiplicación empiezan con el método de las sumas sucesivas. A fin de que los que no dominan este lenguaje comprendan mejor el desarrollo de la rutina, en lugar de mostrar el listado en ASSEMBLER lo haré en BASIC, intentando simular la ejecución del código máquina. Os ruego que respetéis los números que tiene asignados las líneas.

LISTADO 1

```
10 'RUTINA DE MULTIPLIC
ACION
20 'SIMULADA EN BASIC.
30 '____________________
______
40 'HL=HL*DE
50 '____________________
______
60 '
70 INPUT"MULTIPLICANDO"
;MO
80 INPUT"MULTIPLICADOR"
;MR
90 GOSUB 100:GOSUB310:E
ND
100 DE=MO:HL=MR:TIME=0
110 IF HL=0 OR DE=0 THE
N HL=0:RETURN
130 BC=HL
140 HL=0
150 HL=HL+DE
160 BC=BC-1
170 IF BC THEN 150
180 RETURN
310 PRINT"RESULTADO:"HL
"TIEMPO"TIME/50:RETURN
```

El funcionamiento del programa es sencillo. En síntesis, se trata de sumar a HL el contenido del multiplicando tanta veces como el valor del multiplicador. La cosa funciona, pero tiene un gran defecto: cuando el multiplicador es muy grande el bucle de las líneas 150 a 170 se repite un gran número de veces y el tiempo de ejecución se alarga.

Una mejora: dado que la multiplicación tiene la propiedad conmutativa, es posible comparar los dos factores y elegir como multiplicador el más pequeño. Esto se consigue con una línea como esta:

120 IF DE<HL THEN SWAP HL,DE

Aun con la pequeña mejora anterior, el programa dista mucho de ser óptimo, ya que cuando ámbos factores son grandes las esperas se hacen eternas, debido, como he dicho, a que el bucle ha de repetirse un número considerable de veces.

Pasemos ahora a desarrollar el algoritmo para la multiplicación que se enseña en los manuales escolares. Todos sabemos que consiste en ir multiplicando cada una de las cifras parciales del multiplicador por el multiplicando (ver figura 1); para ir desplazando a su vez los distintos resultados un lugar a la izquierda. Sólo resta obtener la suma total y la multiplicación queda resuelta. No obstante, dado que el Z80 ni tan siquiera es

```
  14
x 13
---
  42
 14    figura 1.
----   Multiplicación
 182   en decimal.
```

```
    1110   (=14)
  x 1101   (=13)
  -----
    1110
   0000figura 2.
  1110 Multiplicación
 1110  en binario.
 --------
10110110  (=182)
```

capaz de hacer las multiplicaciones parciales, es imposible copiar exactamente el método tradicional. Observad, empero, que el microprocesador sí puede multiplicar por cero o por uno, en el primer caso el resultado será cero y en el segundo el propio multiplicando. Por consiguiente, basta que cada una de las cifras de los factores no sea mayor que uno para que la multiplicación sea factible. Naturalmente habréis intuido que la única notación que no incluye dígitos mayores de uno es la binaria.

Multiplicar números en binario tal vez os pueda parecer un tanto enrevesado, pero, si se piensa, en más fácil que la forma tradicional en base diez (ver figura 2).

A continuación se incluyen unas líneas que implantan la nueva rutina de multiplicación. Al igual que la primera, está escrita de forma que simule la ejecución en código máquina.



LISTADO 2

```
190 DE=MO:HL=MR:TIME=0
200 B=16
210 AC=HL
220 HL=0
230 IF AC/2=INT(AC/2) T
HEN CARRY=0 ELSE CARRY=
1
240 AC=INT(AC/2)
250 IF CARRY=0 THEN 270
260 HL=HL+DE
270 DE=DE+DE
280 B=B-1
290 IF B THEN 230
300 RETURN
```

En este punto, sólo resta modificar la línea 90 para que el programa se encargue de mostrar los resultados de ambas rutinas, así como los tiempos empleados en el proceso. La nueva línea debe quedar así:

90 GOSUB100:GOSUB310:
GOSUB190:GOSUB310:END

Experimentad ahora con valores pequeños e id aumentándolos progresivamente. Cuando ambos factores son del orden de unos pocos millares, la espera, en el primer método, puede alargarse a minutos, mientras que el segundo siempre invierte una cantidad de tiempo parecida en obtener la multiplicación: alrededor de 4 segundos. Quizá objetéis que para factores pequeños es mejor la primera rutina, pero esto no es del todo cierto. Tened presente que el método de las multiplicaciones parciales se basa en un manejo detallado de los bits del multiplicador y es en este detalle concreto donde el BASIC no tiene nada qué hacer contra el C.M., debido a las facilidades que da el Z80 para tratar bits. Así pues, la segunda rutina gana mucho en eficacia cuando se ejecuta realmente en C.M.

MULTIPLICACION EN ASSEMBLER

Conocéis ya el algoritmo optimizado de la multiplicación, pasemos ahora a comprobar toda su potencia con el ASSEMBLER:

LISTADO 3

```
1               ORG  40000
2               LD   HL,(39994)
3               LD   DE,(39996)
4               CALL MULT
5               LD   (39998),HL
6               RET
7  MULT:        LD   B,16
8               LD   C,H
9               LD   A,L
10              LD   HL,0
11 BUCLE:       SRL  C
12              RRA
13              JR   NC,BITO
14              ADD  HL,DE
15 BITO:        EX   DE,HL
16              ADD  HL,HL
17              EX   DE,HL
18              DJNZ BUCLE
19              RET
```

Las líneas 1 a 6 no forma parte de la rutina de multiplicación propiamente dicha.

La línea 1 se encarga de colocar el código fuente a partir de la dirección 40000.

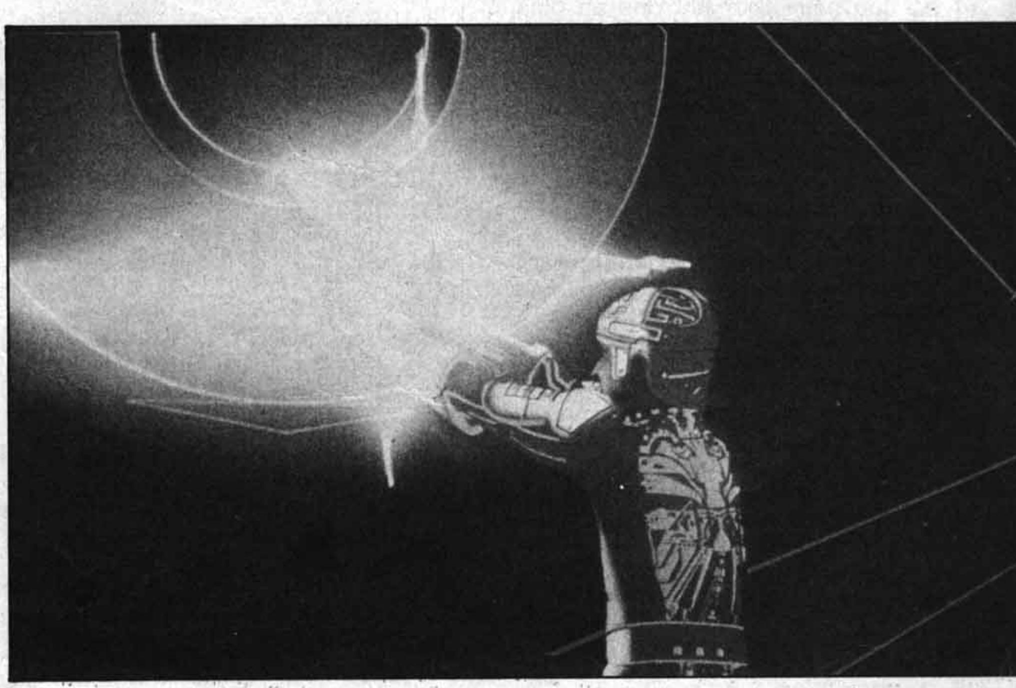
Las líneas 2 y 3 sirve para colocar el multiplicador y el multiplicando en las posiciones 39994 y 39996.

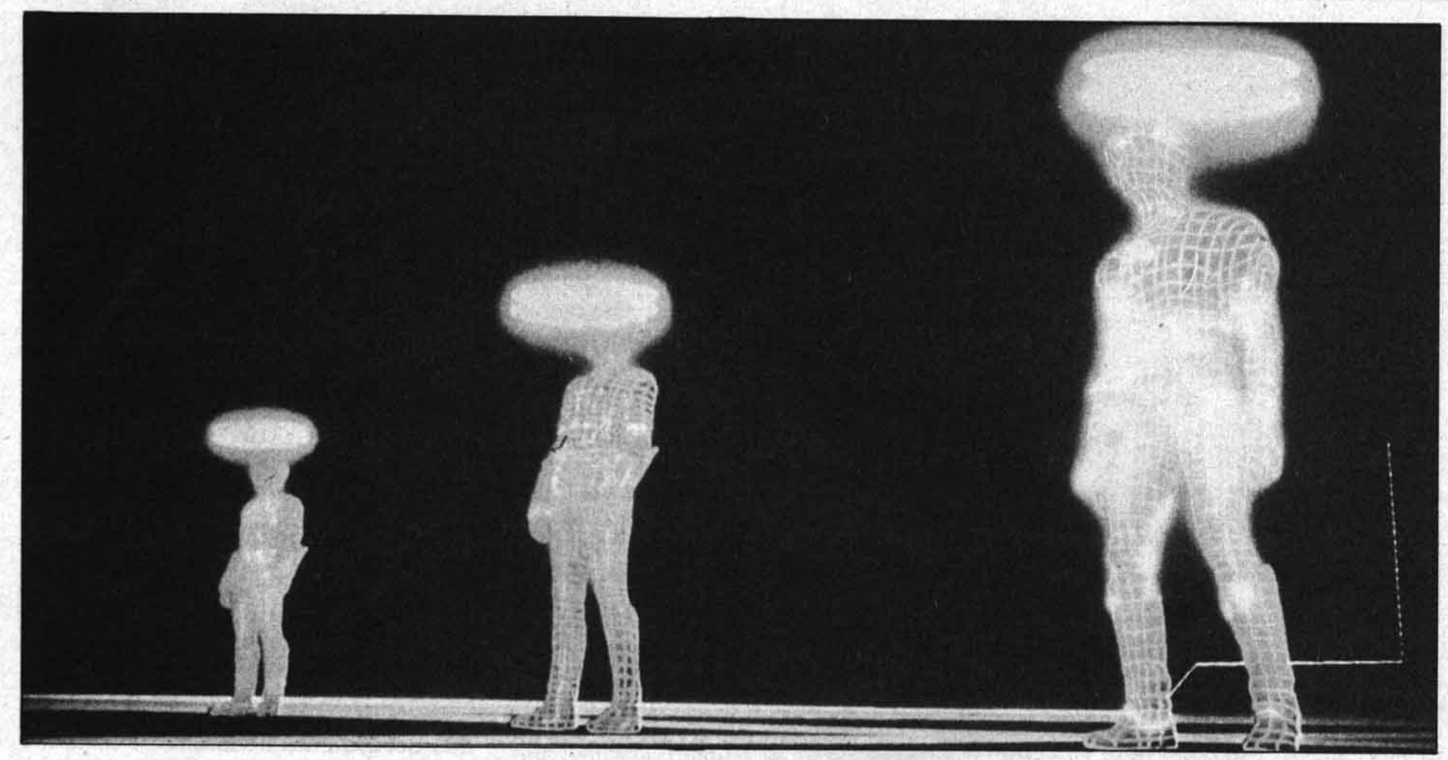
La línea 4 hace una llamada al algoritmo de multiplicación y la línea 5 sirve para poner el resultado en la dirección 39998.

La línea 7 prepara un bucle contador, que servirá para determinar el número de productos parciales. Este número es la cantidad máxima de bits del multiplicador; 16, en este caso.

Las líneas 8 y 9 cargan el contenido del multiplicador original (HL) en los registros A y C, a fin de desocupar el registro doble HL y usarlo para almacenar los productos parciales, borrándolo previamente (línea 10). El motivo de todo esto es que HL es el único registro capaz de hacer sumas de 16 bits (salvo los registros de índice IX e IY).

Las líneas 11 y 12 desplazan el multiplicador un lugar a la derecha, haciendo que el bit sobrante se copie en el CARRY. Así, si el CARRY está encendido, el programa sumará un nuevo producto parcial, en otro caso, sumará cero, es





decir: no sumará: El estado del CARRY se comprueba en la línea 13 con la instrucción JR NC (salta a la línea 15 si el CARRY está apagado).

La línea 14, como ya he dicho, es la encargada de sumar un nuevo producto parcial.

La línea 15 intercambia el contenido de los registros dobles HL y DE, para que la línea 16 duplique el valor de HL y la línea 17 vuelva a conmutar los dos registros. Como veis, se trata, simplemente, de multiplicar DE por 2. Puesto que este registro no es capaz de sumar, debe ser momentáneamente pasado a HL, de ahí la necesidad de la operación de intercambio.

El motivo por el cual DE es duplicado es que multiplicar por dos en binario equivale a desplazar todos los bits un lugar a la izquierda. Así, se realiza el desplazamiento del multiplicando de la misma forma que lo harías tú, poniendo los productos parciales en «escalera».

La línea 18 sirve para comprobar el estado del contador. DJNZ empieza decrementando el registro B, para saltar al inicio del bucle si éste no es cero. Funciona de manera parecida a la instrucción NEXT del BASIC, pero disminuyendo el valor en lugar de aumentarlo.

El siguiente programa incorpora un cargador de líneas DATA. Con él podrás poner la rutina en la memoria y probar su funcionamiento. Debes tener presente que esta rutina sólo admite resultados menores que 2,16. Si la multiplicación es mayor, se entregará un número que será el resto de la división entre el resultado correcto y 2,16.

LISTADO 4

```
10 GOSUB80
20 INPUT"MULTIPLICANDO
";MO
30 INPUT"MULTIPLICADOR
";MR
40 POKE39995!,MO/256:P
OKE39994!,MO-256*INT(M
O/256)
50 POKE39997!,MR/256:P
OKE39996!,MR-256*INT(M
O/256)
60 DEFUSR=40000!:A=USR
(0)
70 PRINT"RESULTADO:"PE
EK(39999!)*256+PEEK(39
998!):END
80 FORX=40000!TO40032!
:READV$
90 POKEX,VAL("&H"+V$)
100 S=S+VAL("&H"+V$):N
EXT
110 IFS<>3286THENBEEP:
CLS:PRINT"HAY UN ERROR
 EN LAS DATAS"
120 RETURN
130 DATA2A,3A,9C,ED,5B
,3C,9C,CD,4E,9C,22,3E,
9C,C9,06,10,4C,7D,21,0
0,00,CB,39,1F,30,01,19
,EB,29,EB,10,F5,C9
```

CONCLUSIONES

Espero haber hecho lo bastante como para tentarte a conseguir pequeños algoritmos (o a construirlos), para incluirlos en tus programas.

Cuando los microordenadores no disfrutaban de los lujos del BASIC, los fabricantes solían incorporar aplicaciones que ayudaran a programar en ASSEMBLER, incluyendo rutinas aritméticas (en binario, BCD y coma flotante), rutinas de búsqueda, de ordenación, etcétera. Desgraciadamente, hoy por hoy no existe algo parecido para los MSX. Hay, sin embargo, bastante literatura sobre el Z80, aunque todo lo bueno está editado, cómo no, en inglés.

Tal vez creas que programar en ASSEMBLER del Z80 es muy complicado. Por si te sirve de consuelo, te diré que es uno de los microprocesadores más difíciles de tratar. Sus registros están muy especializados y cada uno de ellos sirve mejor a una causa concreta. Además, tiene 696 instrucciones diferentes que no hacen agradable la programación a los principiantes. Hay que tener presente que su aparición comercial data de 1976, lo que, en un mercado tan inquieto, es mucho tiempo.

En la actualidad, se especula con que la tercera generación de ordenadores MSX incorporará un micro 68000 y su compatibilidad se mantendrá gracias a un emulador o a una tarjeta que contendrá un Z80. El 68000, implantado hoy en ordenadores como el Appel Macintosh, el Atari 520ST o el Commodore Amiga; es un microprocesador pensado para programar con comodidad. Permite trabajar con muchos acumuladores y modos de direccionamiento y sus registros pueden contener datos de 8, 16 o 32 bits. Basta decir que es capaz de hacer multiplicaciones o divisiones de 32 bits, con o sin signo, en un solo paso.

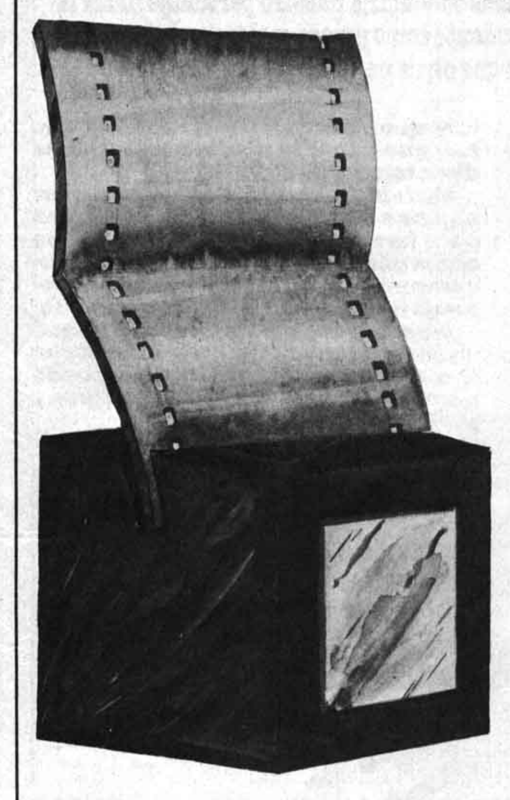
En fin, hasta que la tecnología nos preste su auxilio, habrá que seguir pensando cómo obtener buenos resultados con los medios disponibles.

POR JOAQUIN LOPEZ

2.º GRAN CONCURSO PROGRAMA DEL AÑO



CREA Y ENVIANOS TU PROGRAMA. HAY PREMIOS PARA TI Y PARA LOS QUE TE VOTEN. CADA MES PUBLICAREMOS A MAS DE UN GANADOR QUE OPTARA POR EL «LISTADO DE ORO» Y UNA FABULOSA UNIDAD DE DISCO.



BASES

1– Podrán participar todos nuestros lectores cualquiera sea su edad, con uno o más programas.

2– Los programas se clasificarán en tres categorías:
A- Educativos
B- Gestión
C- Entretenimientos

3– Los programas, sin excepción, deberán ser remitidos grabados en cassette virgen, debidamente protegida dentro de su estuche plástico en el que se insertará el cupón-etiqueta que aparece en esta misma página, debidamente rellenado.

4– No entrarán en concurso aquellos programas plagiados o ya publicados en otras publicaciones nacionales o extranjeras.

5– Junto a los programas se incluirán en hoja aparte las instrucciones correspondientes, detalle de las variables, ampliaciones o mejoras posibles y todos aquellos comentarios que el autor considere de interés.

6– Todos los programas han de estar estructurados de modo claro, separando con REM los distintos apartados del mismo.

PREMIOS

7– SUPER JUEGOS EXTRA MSX otorgará los siguientes premios:
AL PROGRAMA EXTRA MSX DEL AÑO
«EL LISTADO DE ORO»
Una Unidad de disco
valorada en más de 80.000 ptas.

8– Los programas seleccionados por nuestro Departamento de Programación y publicados en cada número de nuestra revista recibirán los siguientes premios en metálico:
Programa Educativo 10.000 pts.
Programa de Gestión 10.000 pts.
Programa de Entretenimiento 6.000 pts.

9– SUPER JUEGOS EXTRA MSX se reserva el derecho de publicar fuera de concurso aquellos programas de reducidas dimensiones que sean de interés, premiando a sus autores.

FALLO Y JURADO

10– Nuestro Departamento de Programación analizará todos los programas recibidos y hará la primera selección, de la que saldrán los programas que publiquemos en cada número de S.J. EXTRA MSX.

11– Los programas recibidos no se devolverán, salvo que el autor lo requiera expresamente.

12– La elección del PROGRAMA MSX EXTRA DEL AÑO se hará por votación de nuestros lectores a través de un boletín que se publicará en el mes de octubre de 1986.

13– El plazo de entrega de los programas finaliza el 14 de noviembre de 1986.

13– El fallo se dará a conocer en el número del mes de enero de 1987, entregándose los premios el mismo mes.

REMITIR A:
CONCURSO EXTRA MSX
Roca i Batlle, 10-12
bajos
08023 Barcelona

MSX EXTRA

PROGRAMA .... N.º

NOMBRE DEL PROGRAMA ....

CATEGORIA
PARA .... K
INSTRUCCION DE CARGA

AUTOR:
EDAD:
CALLE: .... N.º ....
CIUDAD .... DP .... TEL.: ....
N.º DE RECEPCION

OGROS

Por Jesús Tubío Fernández

En este divertido juego nos encontramos en la caverna de unos terribles ogros. Debemos conducir a nuestro personaje hacia la salida pasando por diferentes cavernas. ¡Atención! pasar a algunas cavernas no es tan evidente como parece, y algunos de los objetos que aparezcan en las habitaciones serán útiles, mientras que otros no.

Instrucciones. En este juego hay que llevar al muñeco hasta la salida. Para ello sólo hay una forma, aunque las instrucciones que lleva el juego dicen que hay que coger las cosas una no se debe coger para poder llegar al final, si no se hace de la forma que explicare a continuación no se podrá pasar al otro lado de la catarata que es por donde se llega a la salida.

Forma de salir: para ello no hay que coger el puñal que hay en la pantalla tercera al no cogerlo y pasar a la siguiente pantalla éste se verá en la catarata lo único que hay que hacer es tocarlo para que pasemos al final.

Forma de coger el diamante: éste se encuentra en la segunda pantalla y aunque intentemos llegar a él por el puente nos caeremos al agua para poder cogerlo hay que ir a la pantalla número 7 que es la habitación y coger la cruz al pasar de pantalla llegaremos a la segunda y ya sí podremos coger el diamante.

Forma de entrar en la casa: en la pantalla quinta la que representa el jardín para poder entrar en la casa no hay que acercarse a muro sino que hay que coger la piedra y pasaremos directamente.

Variables. Son fáciles de entender algunas son: A$, B$, C$, D$ sirven para hacer el muñeco. E$, G$, H$ se encargan de la música. T$, T2$ sirven para hacer las estalactitas y el fuego de la antorcha respectivamente.

Ampliaciones y mejoras. Se puede mejorar el movimiento y el muñeco que no está del todo bien. Se le puede añadir más pantallas y más peligros y problemas.

```
100 GOTO 3000
104 ' AAAAAJ.T.F.AAAAA
105 CLS
106 COLOR ,1,13
110 SCREEN 2,2
114 T=0
119 '*****SPRITE$*****
120 A$=CHR$(&H0)+CHR$(&H0)+CHR$(&H0
)+CHR$(&H0)+CHR$(&H1)+CHR$(&H1)+CHR
$(&H0)+CHR$(&H3)
130 B$=CHR$(&H3)+CHR$(&H3)+CHR$(&H3
)+CHR$(&H3)+CHR$(&H1)+CHR$(&H1)+CHR
$(&H1)+CHR$(&H1)
140 C$=CHR$(&H0)+CHR$(&HC0)+CHR$(&H
A0)+CHR$(&HC0)+CHR$(&H0)+CHR$(&HC0)
+CHR$(&H80)+CHR$(&HC0)
150 D$=CHR$(&HC0)+CHR$(&HC0)+CHR$(&
HC0)+CHR$(&H80)+CHR$(&H80)+CHR$(&H8
0)+CHR$(&H80)+CHR$(&HC0)
160 SPRITE$(1)=A$+B$+C$+D$
165 SPRITE$(3)=CHR$(&H18)+CHR$(&H18
)+CHR$(&H3C)+CHR$(&H7E)+CHR$(&H7E)+
CHR$(&H7E)+CHR$(&H7E)+CHR$(&H7E)
166 SPRITE$(4)=CHR$(&H18)+CHR$(&H18
)+CHR$(&H7E)+CHR$(&H3C)+CHR$(&H3C)+
CHR$(&H3C)+CHR$(&H3C)+CHR$(&H18)
167 SPRITE$(5)=CHR$(&H3C)+CHR$(&H7E
)+CHR$(&HFF)+CHR$(&H7E)+CHR$(&H3C)+
CHR$(&H18)+CHR$(&H0)+CHR$(&H0)
168 SPRITE$(6)=CHR$(&H18)+CHR$(&H18
)+CHR$(&H7E)+CHR$(&H18)+CHR$(&H18)+
CHR$(&H18)+CHR$(&H18)+CHR$(&H18)
169 SPRITE$(7)=CHR$(&HFF)+CHR$(&H7E
)+CHR$(&H3C)+CHR$(&H18)+CHR$(&H18)+
CHR$(&H18)+CHR$(&H18)+CHR$(&H7E)
190 SPRITE$(8)=CHR$(&H7E)+CHR$(&H7E
)+CHR$(&H3C)+CHR$(&H3C)+CHR$(&H18)+
CHR$(&H18)+CHR$(&H18)+CHR$(&H18)
195 SPRITE$(9)=CHR$(&H3C)+CHR$(&H7E
)+CHR$(&HFF)+CHR$(&HFF)+CHR$(&HFE)+
CHR$(&HFF)+CHR$(&H7E)+CHR$(&H3C)
196 T$="C13BM15,0M23,30M35,15M40,26
M46,17M55,35M63,13M77,50M85,30M100,
49M120,22M140,30M158,22M190,39M210,
20M217,60M230,15M246,0"
200 T2$="C6BM121,85M120,83M122,79M1
21,81M121,79M121,76M126,80M127,83M1
26,85M121,85"
400 ' ***1 PANTALLA
405 CLS
406 PUT SPRITE 5,(235,93),1,5
410 LINE (0,120)-(255,140),6,BF
412 LINE (14,20)-(18,22),8,BF
413 LINE (20,20)-(24,22),8,BF
414 LINE (18,24)-(22,26),8,BF
415 LINE (24,24)-(28,26),8,BF
430 PUT SPRITE 8,(120,85),14,8
440 DRAW T2$
445 PAINT (122,83),6,6
450 LINE (80,85)-(95,120),12,BF
460 LINE (85,90)-(89,94),1,BF
470 LINE(85,92)-(89,92),12
480 LINE (87,90)-(87,94),12
490 LINE (150,85)-(165,120),12,BF
500 LINE (155,90)-(159,94),1,BF
510 LINE (155,92)-(159,92),12
520 LINE (157,90)-(157,94),12
530 E$="T130S12V10O6M2500CDECDEFEDG
```

PROGRAMAS

```
FEC"
531 PLAY E$:PLAY E$
600 X=10
605 Y=105
610 PUT SPRITE 1,(X,Y),15,1
620 D=STICK(0)
630 IF D=0 THEN PUT SPRITE 1,(X,Y),
15,1
640 IF D=3 THEN X=X+1:IF X>255 THEN
 X=0:GOTO 800
650 IF D=7 THEN X=X-1:IF X<0 THEN X
=120:GOTO 1000
660 GOTO 610
800 ' ***2 PANTALLA***
801 ON SPRITE GOSUB870
802 SPRITE ON
805 CLS:PUT SPRITE 8,(120,85),1,8
806 PUT SPRITE 6,(65,50),1,6
810 LINE (0,100)-(180,120),6,BF
815 LINE (225,100)-(255,120),6,BF
820 DRAW T$
825 PAINT (60,15),13,13
830 LINE (0,130)-(255,191),5,BF
835 LINE(0,187)-(255,191),6,BF
840 PUTSPRITE 5,(235,93),7,5
849 LINE (180,100)-(225,102),6,BF
850 X=10
853 Y=86
855 PUT SPRITE 1,(X,Y),15,1
856 D=STICK(0)
859 IF D=0 THEN PUT SPRITE 1,(X,Y),
15,1
861 IF D=3 THEN X=X+1:IF X>180 AND
T=0THEN Y=140 ELSE Y=86
862 IF Y=140ANDX=185 THEN FOR R=1 T
O700:NEXT:GOTO 3200
863 IF D=7 THEN X=X-1:IF X<0 THEN X
=0:GOTO400
864 PUT SPRITE 1,(X,Y),15,1
865 GOTO 855
870 IF STRIG(0)=-1 THEN PUT SPRITE
5,(-10,0),1,5:GOTO 1000
880 RETURN
890 SPRITE OFF
1000 '*****3 PANTALLA*****
1005 CLS
1006 ON SPRITE GOSUB 1080
1007 SPRITE ON
1010 LINE (0,120)-(255,130),6,BF
1011 PUT SPRITE 8,(120,85),14,8
1015 DRAW T$
1020 PAINT (60,15),13,13
1025 DRAW T2$
1027 PAINT (122,83),6,6
1028 LINE (180,127)-(185,191),12,BF
1031 LINE (185,140)-(255,191),5,BF
1033 LINE (185,185)-(255,191),6,BF
1040 PUT SPRITE 4,(100,106),14,4
1045 X=10
1050 Y=105
1053 PUT SPRITE1,(X,Y),15,1
1055 D=STICK(0)
1057 IF D=0 THEN PUT SPRITE1,(X,Y),
15,1
1060 IF D=3 THEN X=X+1:IF X>255 THE
N 1200
1070 IF D=7 THEN X=X-1:IF X<0 THEN
X=0
1073 GOTO 1053
1080 IF STRIG(0)=-1 THEN PUT SPRITE
 4,(-20,10),1,4:R=1
1090 RETURN
1095 SPRITE OFF
1100 GOTO 1100
1200 ' *****4 PANTALLA******
1202 OPEN "grp:"AS#1
1203 CLS:PUT SPRITE 8,(120,85),5,8
1204 ON SPRITE GOSUB 1280
1205 SPRITE ON
1208 LINE (0,120)-(100,130),6,BF
1210 LINE (155,120)-(255,130),6,BF
1220 LINE (0,140)-(255,191),5,BF
1225 DRAW"C5BM107,0M99,140M156,140M
148,0M107,0"
1227 PAINT (120,100),5,5
1230 LINE (100,120)-(155,120),6
1232 LINE (180,80)-(183,80),8
1234 COLOR 8
1235 PRINT#1,"EXIT▶"
1240 LINE(0,186)-(255,191),6,BF
1245 DRAW T$
1247 PAINT (60,15),13,13
1248 PUT SPRITE 7,(220,115),7,7
1249 CLOSE
1250 X=10
1255 Y=105
1257 PUT SPRITE 1,(X,Y),15,1
1258 D=STICK(0)
1260 IF D=0 THEN PUT SPRITE 1,(X,Y)
,15,1
1264 IF D=3 THEN X=X+1:IF X>=100 TH
EN Y=143ELSE Y=105
1265 IF Y=143 AND X=103 THEN FOR R=
1TO 750:NEXT:GOTO 3200
1266 IF D=7 THEN X=X-1:IF X<0 THEN
1600
1270 GOTO 1257
1280 IF STRIG(0)=-1 THEN PUT SPRITE
7,(-30,0),7,7
1283 SPRITE OFF
```

PROGRAMAS

```
1285 GOTO 1400
1399 '****ULTIMA PANTALLA****
1400 CLS:COLOR ,,1:PUT SPRITE7,(-5,
Ø),1,7
1401 PUT SPRITE 8,(-15.Ø),1,8
1402 PUT SPRITE 1,(1Ø,132),15,1
1403 PUT SPRITE 4,(-6,Ø),1,4
1406 C=C+1
1407 F=RND(1)*255
1408 E=RND(1)*15Ø
141Ø PRESET(F.E),15
1412 IF C=5Ø THEN 142ØELSE 1406
142Ø CIRCLE (4Ø,191).8Ø,15,,,6.12
143Ø PUT SPRITE 1,(35,173),15,1
1432 PLAY"S3M11000T12004CECFCEFGFEC
ECFCEFGFECDCEDEFEDC"
144Ø OPEN "grp:"AS#1
1445 COLOR 8
1446 LINE(1Ø,1ØØ)-(1Ø,1Ø2),8
145Ø PRINT#1,"ERES  LIBRE"
15ØØ GOTO 15ØØ
16ØØ '******5 PANTALLA****
16Ø3 PUT SPRITE 5,(-1Ø,Ø),1,5
16Ø4 PUT SPRITE 8,(-25,85),14,8
16Ø5 CLS:PUT SPRITE7,(-23,Ø),7,7
16Ø6 OPEN "grp:"AS#1
16Ø7 ON SPRITE GOSUB 168Ø
16Ø8 SPRITE ON
161Ø LINE (Ø,14Ø)-(255,191),6,BF
1615 LINE(14Ø,13Ø)-(19Ø,14Ø),7,BF
1616 LINE (14Ø,13Ø)-(19Ø,14Ø),13,B
1617 LINE(161,1ØØ)-(17Ø,13Ø),13,BF
162Ø LINE (156,1ØØ)-(176,1ØØ),13
1623 LINE (166,1ØØ)-(158,8Ø),7
1625 LINE (166,1ØØ)-(174,8Ø),7
1628 LINE (166,1ØØ)-(166,98),13
163Ø PUT SPRITE 9,(1ØØ,134),8,9
1632 LINE (Ø,14Ø)-(255,145),3,BF
1634 LINE (22Ø,Ø)-(255,14Ø),14,BF
1635 LINE (22Ø,1ØØ)-(21Ø,1Ø4),8,BF
1636 PRESET(18Ø,6Ø),1
1637 G$="T15ØS11M1ØØØØ4DECFDECGFED
CDEC."
1638 PLAY G$:PLAY G$
1646 PRINT#1,"CASA▶"
1647 LINE (23Ø,8Ø)-(239,86),8,BF
1648 LINE(237,78)-(246,72),8,BF
1649 CLOSE
1656 X=5:Y=124
1657 PUT SPRITE1,(X,Y),15,1
1658 D=STICK(Ø)
1659 IF D=Ø THEN PUT SPRITE 1,(X,Y)
,15,1
166Ø IF D=3 THEN X=X+1:IF X>21Ø THE
N X=21Ø
1664 IF D=7 THEN X=X-1:IF X<Ø THEN
X=Ø
1668 GOTO 1657
168Ø IF STRIG(Ø)=-1 THEN PUT SPRITE
9,(-24,2),8,9:GOTO 18ØØ
169Ø RETURN
18ØØ '*******6 PANTALLA*******
181Ø CLS:PUT SPRITE 1,(-2Ø,Ø),1,1
1812 FOR E=1 TO 12ØØ:NEXT
1814 ON SPRITE GOSUB 19ØØ
1815 SPRITE ON
1817 LINE (Ø,12Ø)-(255,191),6,BF
1818 LINE (6Ø,9Ø)-(64,12Ø),14,BF
1819 LINE (18Ø,9Ø)-(184,12Ø),14,BF
182Ø LINE (6Ø,1Ø5)-(18Ø,1Ø5),14,BF
1821 LINE (57,85)-(187,9Ø),13,BF
1822 LINE (22Ø,6Ø)-(22Ø,12Ø),13,BF
1823 LINE (22Ø,9Ø)-(2ØØ,95),13,BF
1824 LINE (2Ø2,95)-(2Ø4,12Ø),13,BF
1825 DRAW"C12BM2Ø,ØM2,7ØM15,12ØMØ,1
2ØMØ,ØM2Ø,Ø"
1826 PAINT(8,1ØØ),12,12
183Ø PUT SPRITE 7,(16Ø,77),7,7
1875 LINE (Ø,69)-(2,71),1Ø,BF
1876 PUT SPRITE 3,(1ØØ,11Ø),7,3
188Ø X=18:Y=1Ø4
1883 PUT SPRITE 1,(X,Y),15,1
1885 D=STICK(Ø)
1888 IF D=Ø THEN PUT SPRITE 1,(X,Y)
,15,1
189Ø IF D=3 THEN X=X+1:IF X>255 THE
N X=Ø:GOTO 2ØØØ
1893 IF D=7 THEN X=X-1 :IF X<Ø THEN
 X=255
1895 GOTO 1883
19ØØ IF STRIG(Ø)=-1 THEN PUT SPRITE
 3,(-3Ø,Ø),7,3
191Ø RETURN
2ØØØ '********7 PANTALLA******
2Ø18 ON SPRITE GOSUB 2165
2Ø19 SPRITE ON
2Ø2Ø CLS:PUT SPRITE 7,(4Ø,8ØØ),1,7
2Ø25 LINE (Ø,14Ø)-(255,191),6,BF
2Ø3Ø LINE (1ØØ,1Ø5)-(1Ø2,14Ø),14,BF
2Ø35 LINE (18Ø,98)-(182,14Ø),14,BF
2Ø4Ø LINE (1Ø2,115)-(18Ø,13Ø),11,BF
2Ø45 CIRCLE(175,112),8,15,,,.23
2Ø5Ø PAINT (175,112),15,15
2Ø55 LINE (5Ø,1Ø5)-(7Ø,135),13,BF
2Ø6Ø LINE (5Ø,135)-(5Ø,14Ø),13
2Ø65 LINE (7Ø,135)-(7Ø,14Ø),13
2Ø7Ø CIRCLE (6Ø,81),24,7,,,2.7
2Ø75 PAINT (6Ø,81),7,7
2Ø8Ø LINE (53,1Ø5)-(53,97),13
2Ø85 LINE (67,1Ø5)-(67,97),13
```




```
2Ø9Ø CIRCLE(6Ø,11Ø),1,1,,,1.4
2Ø95 PAINT (6Ø,11Ø),1,1
21ØØ CIRCLE(6Ø,12Ø),1,1,,,1.4
21Ø5 PAINT (6Ø,12Ø),1,1
211Ø CIRCLE(6Ø,13Ø),1,1,,,1.4
2115 PAINT (6Ø,13Ø),1,1
212Ø LINE (5Ø,1Ø5)-(7Ø,115),1,B
2125 LINE (5Ø,115)-(7Ø,125),1,B
213Ø LINE (5Ø,125)-(7Ø,134),1,B
2135 PUT SPRITE 6,(11Ø,134),8,6
2137 G$="T14ØV9DECEDECFEDFECDDECEDE
CFEFCDE"
2138 H$="T14ØV6O5GFEGFEGFEGFEGFEGFE
GFE"
214Ø PLAY G$,H$
2145 X=7:Y=124
2147 PUT SPRITE 1,(X,Y),15,1
215Ø D=STICK(Ø)
2154 IF D=3 THEN X=X+1:IF X=255 THE
N8ØØ
2157 IF D=7 THEN X=X-1:IF X=Ø THEN
X=255
216Ø GOTO 2147
2165 IF STRIG(Ø)=-1 THEN PUT SPRITE
 6,(-6Ø,Ø),1,6:T=1
2168 RETURN
22ØØ GOTO 22ØØ
2999 '******PRINCIPIO*****
3ØØØ OPEN"grp:" AS#1
3ØØ2 SCREEN 3
3ØØ3 PRESET(1Ø,1ØØ),13
3ØØ5 COLOR 13,12
3Ø1Ø PRINT#1,"OGROS"
3Ø12 FOR R=1TO 5ØØ:NEXT
3Ø15 COLOR 13
3Ø2Ø PRINT#1,"██████████████
████████████████████████████
██████  "
3Ø3Ø SCREEN Ø
3Ø35 COLOR 1,15
3Ø4Ø LOCATE 8,Ø
3Ø5Ø PRINT"==INSTRUCCIONES=="
3Ø6Ø LOCATE 2,1Ø
3Ø7Ø PRINT"TE HAS ESCAPADO DE LA CE
LDA DONDE ESTABAS PRISIONERO.PARA P
ODER SALIR DEBERAS COGER ALGUNOS OB
JETOS RAPIDAMENTE Y CON CUIDADO.UNO
 DE LOS OBJETOS QUE DEBES COGER ES
EL GRAN DIAMANTE."
3Ø8Ø PRINT"DEBES IR DEPRISA PORQUE
LOS OGROS TE QUIEREN COMO CENA.ANIM
O Y SUERTE"
3Ø85 CLOSE
3Ø9Ø INPUT"QUIERES EMPEZAR (S/N)";A
$
31ØØ IF A$="S" THEN 1Ø5 ELSE 31ØØ
31Ø5 RETURN
32ØØ '*******MUERTE*********
32Ø3 FOR R=1 TO 5ØØ:NEXT
32Ø5 CLS:PUT SPRITE 1,(X,Y),1,1
32Ø7 COLOR 15
321Ø SCREENØ
3215 LOCATE 5,15
322Ø PRINT"HAS MUERTO AHOGADO.LO SI
ENTO"
3225 END
```

TEST DE LISTADO

Para utilizar el Test de Listados que ofrecemos al final de cada programa, recordamos que previamente hay que cargar en el ordenador el Programa correspondiente aparecido en nuestro número 10, página 29.

1ØØ - 9Ø	169 - 11	45Ø - 78	65Ø -213	853 -173	1Ø1Ø -2Ø1	1Ø7Ø -163
1Ø4 - 58	19Ø -139	46Ø - 47	66Ø -251	855 -11Ø	1Ø11 -181	1Ø73 -184
1Ø5 -159	195 - 74	47Ø -132	8ØØ - 58	856 - 54	1Ø15 - 54	1Ø8Ø -2Ø6
1Ø6 - 67	196 -211	48Ø -132	8Ø1 - 96	859 - 23	1Ø2Ø - 53	1Ø9Ø -142
11Ø - 23	2ØØ -1Ø1	49Ø -218	8Ø2 - 92	861 -171	1Ø25 -1Ø4	1Ø95 -178
114 - 84	4ØØ - 58	5ØØ -187	8Ø5 -131	862 -239	1Ø27 -173	11ØØ -231
119 - 58	4Ø5 -159	51Ø - 16	8Ø6 - 76	863 - 5	1Ø28 -125	12ØØ - 58
12Ø - 29	4Ø6 - 31	52Ø - 16	81Ø - 96	864 -11Ø	1Ø31 -2Ø8	12Ø2 - 64
13Ø - 33	41Ø -211	53Ø - 6	815 -138	865 -241	1Ø33 -254	12Ø3 -135
14Ø - 31	412 - 26	531 -142	82Ø - 54	87Ø -237	1Ø4Ø -174	12Ø4 -247
15Ø -224	413 - 38	6ØØ - 96	825 - 53	88Ø -142	1Ø45 - 96	12Ø5 - 92
16Ø -17Ø	414 - 42	6Ø5 -192	83Ø - 15	89Ø -178	1Ø5Ø -192	12Ø8 - 46
165 - 32	415 - 54	61Ø -11Ø	835 - 73	1ØØØ - 58	1Ø53 -11Ø	121Ø - 98
166 - 33	43Ø -181	62Ø - 54	84Ø - 37	1ØØ5 -159	1Ø55 - 54	122Ø - 25
167 - 7Ø	44Ø -1Ø4	63Ø - 23	849 - 45	1ØØ6 - 51	1Ø57 - 23	1225 -174
168 - 35	445 -173	64Ø -1Ø9	85Ø - 96	1ØØ7 - 92	1Ø6Ø - 3	1227 -186

123Ø - 9	14Ø3 -2Ø9	162Ø - 51	181Ø -176	191Ø -142	2125 -235	3Ø35 - 25
1232 - 39	14Ø6 -12Ø	1623 - 19	1812 - 46	2ØØØ - 58	213Ø -254	3Ø4Ø - 46
1234 -214	14Ø7 -32	1625 - 35	1814 -1Ø6	2Ø18 -116	2135 -212	3Ø5Ø -126
1235 - 63	14Ø8 -182	1628 - 49	1815 - 92	2Ø19 - 92	2137 - 78	3Ø6Ø - 48
124Ø - 72	141Ø - 21	163Ø -2Ø8	1817 - 6	2Ø2Ø -236	2138 - 86	3Ø7Ø -213
1245 - 54	1412 -227	1632 -233	1818 - 34	2Ø25 - 26	214Ø -196	3Ø8Ø -249
1247 - 53	142Ø - 86	1634 - 61	1819 - 18	2Ø3Ø -147	2145 -1Ø8	3Ø85 -18Ø
1248 - 48	143Ø -171	1635 - 74	182Ø -15Ø	2Ø35 - 44	2147 -11Ø	3Ø9Ø -188
1249 -18Ø	1432 - 64	1636 -14Ø	1821 -118	2Ø4Ø -224	215Ø - 54	31ØØ -199
125Ø - 96	144Ø - 64	1637 - 47	1822 - 63	2Ø45 - 9	2154 - 83	31Ø5 -142
1255 -192	1445 -214	1638 -146	1823 - 48	2Ø5Ø - 13	2157 -159	32ØØ - 58
1257 -11Ø	1446 -25Ø	1646 - 29	1824 - 64	2Ø55 - 59	216Ø - 2	32Ø3 -124
1258 - 54	145Ø -211	1647 - 75	1825 -153	2Ø6Ø -15Ø	2165 -244	32Ø5 - 59
126Ø - 23	15ØØ -12Ø	1648 - 73	1826 - 86	2Ø65 -19Ø	2168 -142	32Ø7 -219
1264 -236	16ØØ - 58	1649 -18Ø	183Ø -2Ø6	2Ø7Ø -132	22ØØ - 55	321Ø -214
1265 -226	16Ø3 -213	1656 -1Ø6	1875 - 98	2Ø75 -111	2999 - 58	3215 - 56
1266 -159	16Ø4 - 72	1657 -11Ø	1876 -171	2Ø8Ø - 83	3ØØØ - 64	322Ø - 27
127Ø -132	16Ø5 -197	1658 - 54	188Ø - 97	2Ø85 -111	3ØØ2 -217	3225 -129
128Ø - 79	16Ø6 - 64	1659 - 23	1883 -11Ø	2Ø9Ø -115	3ØØ3 - 2Ø	
1283 -178	16Ø7 -141	166Ø - 6Ø	1885 - 54	2Ø95 -128	3ØØ5 - 32	
1285 - 2Ø	16Ø8 - 92	1664 -163	1888 - 23	21ØØ -125	3Ø1Ø -192	
1399 - 58	161Ø - 26	1668 - 22	189Ø - 65	21Ø5 -138	3Ø12 -124	
14ØØ - 16	1615 - 39	168Ø - 48	1893 -16Ø	211Ø -135	3Ø15 -217	
14Ø1 -224	1616 -229	169Ø -142	1895 -249	2115 -148	3Ø2Ø -246	TOTAL:
14Ø2 -1Ø5	1617 - 4	18ØØ - 58	19ØØ - 71	212Ø -215	3Ø3Ø -214	31235



MISION IMPOSIBLE

Por César de Pablo Moya

El objetivo de este juego es aprovisionar de combustible a tu base de helicópteros. Para ello deberás conducir tu helicóptero mediante las flechas del cursor hacia la base de aprovisionamiento situada en el ángulo inferior derecho de la pantalla cuidando de no chocar con ningún obstáculo. Cada vez que lleves una nueva carga, aparecerá un obstáculo más.

```
10 '◎◎◎◎◎◎◎◎◎◎◎◎◎◎◎◎◎◎
20 '◎◎ MISION       ◎◎◎
30 '◎◎  IMPOSIBLE  ◎◎◎
40 '◎◎ por CESAR    ◎◎◎
50 '◎◎  DE PABLO    ◎◎◎
60 '◎◎    para      ◎◎◎
70 '◎◎ MSX-EXTRA    ◎◎◎
80 '◎◎◎◎◎◎◎◎◎◎◎◎◎◎◎◎◎◎
90 DATA ,,,f2,2,1f,32,63,7f,3f,92,7
f,,,,,,,,,78,2,87,fa,cØ,8Ø,,,8Ø,,,,:
'HEL<
100 DATA ,,,f,2,1f,32,63,7f,3f,92,7
f,,,,,,,,,8Ø,5,82,fd,cØ,8Ø,,,8Ø,,,,:
'HEL<
110 DATA ,,,1e,4Ø,e1,5f,3,1,,,1,,,,
,,,,4f,4Ø,f8,4c,c6,fe,fc,49,fe,,,,:
'HEL>
120 DATA ,,,3,aØ,41,bf,3,1,,,1,,,,,
,,,fØ,4Ø,f8,4c,c6,fe,fc,49,fe,,,,:'
HEL>
130 DATA ,,,79,1,3,6,c,f,3,4,18,,,,
,,,,3c,,8Ø,cØ,6Ø,eØ,8Ø,4Ø,3Ø,,,,:'H
EL.
140 DATA ,,,7,1,3,6,c,f,3,4,18,,,,,
,,,cØ,,8Ø,cØ,6Ø,eØ,8Ø,4Ø,3Ø,,,,:'HE
L.
150 DATA 34,26,5a,4e,5f,6e,56,e7,3d
,2a,48,67,67,f4,5c,64,3d,42,34,76,5
f,4d,3e,6c,8b,94,23,36,dc,cc,2a,d8:
'EXPLO
160 '◎◎◎ INSTRUCCIONES ◎◎◎
170 SCREEN1,2:WIDTH3Ø:COLOR 15,4,4:
OPEN"grp:"AS#1:KEYOFF
180 FORT=ØTO2STEP2:LOCATE2,T:PRINTS
TRING$(26,CHR$(2Ø3)):NEXT:LOCATE2,1
:PRINT"▒▒   MISION IMPOSIBLE   ▒▒"
190 PRINT:PRINT:PRINT"=ERES EL PILO
TO DE UN HELICOP-TERO. TU MISION ES
```

PROGRAMAS

```
 LLENAR LOS DEPOSITOS DE TU BASE.PA
RA ELLODEBES RECOGER EL COMBUSTIBLE
  DE LA BASE DE APROVISIONAMIEN-TO
ANTES DE QUE SE TE ACABE ELTIEMPO."
210 FORSP%=0TO6:SP$="":FORAA%=1TO32
:READAA$:SP$=SP$+CHR$(VAL("&H"+AA$)
):NEXT:SPRITE$(SP%)=SP$:NEXT
215 PRINT:PRINT:PRINT"    <<<PULSA U
NA TECLA>>>"
220 IFINKEY$=""GOTO220
230 FORT=1TO21:PRINT:NEXT:PRINT" 0.
CURSOR":PRINT" 1.JOYSTICK":FORT=1TO
10:PRINT:NEXT
240 A$=INKEY$:IFA$="1"ORA$="0"THENA
=VAL(A$)ELSE240
250 COLOR3,4,4:SCREEN2
260 '@@@@ DIB. PANTALLA@@@
270 LINE(4,0)-(251,170),2,BF:LINE(8
,2)-(247,169),1,BF
280 DRAW"c2s4a0bm8,159f5e3f3r30e6r2
e6f8e6f4r6e9f5e7f3e9f6e10f8e8f11e5f
15e20f10e9f15e10m247,169m8,169m8,15
9":PAINT(9,162),2
290 DRAW"c10bm20,163u10l5u20r40d313
7d14r5d11r20u10r3d12l26":PAINT(21,1
62),10
300 LINE(26,160)-(39,158),13,BF
310 LINE(216,0)-(247,169),10,BF
320 LINE(216,6)-(220,26),1,BF:LINE(
220,4)-(243,35),1,BF
330 LINE(220,35)-(243,37),13,BF
340 LINE(224,40)-(239,165),15,BF
350 S=RND(-TIME):FORAE=1TO10:H=(RND
(1)*180+10):V=RND(1)*120+5:DRAW"c2b
m=h;,=v;f2e2f3e2fe4f2g5f2g3f2g2h3g5
h3g2h3e5h4e2h3e2f3":PAINT(H,V+1),2:
PAINT(H+8,V+8),2:NEXT
360 PRESET(20,172):PRINT#1,"NAVES";
SPC(11);"CARGAS  0":PRESET(18,182):
PRINT#1,"TIEMPO"
370 FORT=0TO4:PUTSPRITET,(T*20+65,1
67),15,0:NEXT:LINE(70,185)-(229,187
),15,BF
380 '@@@@@@ VARIABLES @@@@@@
390 VI=4:PU=0:FU=230
400 X=225:Y=20:HO=0:VE=0:SC=15
410 SOUND8,16:SOUND9,16:SOUND10,16:
SOUND13,12:SOUND12,4:SOUND6,10:SOUN
D7,255
420 '@@@@@ MOV.CURSOR@@@@
430 ST%=STICK(A):ONST%GOTO450,460,4
70,480,490,500,510,520
440 GOTO530
450 VE=VE-.1:GOTO530
460 VE=VE-.1:HO=HO+.1:GOTO530
470 HO=HO+.1:GOTO530
480 HO=HO+.1:VE=VE+.1:GOTO530
490 VE=VE+.1:GOTO530
500 VE=VE+.1:HO=HO-.1:GOTO530
510 HO=HO-.1:GOTO530
520 HO=HO-.1:VE=VE-.1:GOTO530
530 X=X+HO*4:Y=Y+VE*4
540 IFHO>.1THENFO%=2ELSEIFHO<-.1THE
NFO%=0ELSEFO%=4
550 IFMO%=0THENMO%=1ELSEMO%=0
560 FO%=FO%+MO%:PUTSPRITEVI,(X,Y),S
C,FO%
570 FU=FU-.3:LINE(FU,185)-(FU,187),
4:IFFU<70THEN750ELSEIFFU<100THENIFS
C=13THENSC=15ELSESC=13
580 ONFO%GOTO590,600,600,610,610
590 P1=POINT(X,Y+4):P2=POINT(X+16,Y
+6):P3=POINT(X,Y+12):P4=POINT(X+8,Y
+12):GOTO620
600 P1=POINT(X,Y+6):P2=POINT(X+16,Y
+4):P3=POINT(X+7,Y+12):P4=POINT(X+1
6,Y+12):GOTO620
610 P1=POINT(X+1,Y+4):P2=POINT(X+14
,Y+4):P3=POINT(X+3,Y+12):P4=POINT(X
+12,Y+12):GOTO620
620 IFP1=10ORP1=2ORP2=10ORP2=2ORP3=
10ORP3=2ORP4=10ORP4=2THEN750
630 IFP3=13ORP4=13THEN660
640 GOTO430
650 '@@@@ RECOGIDA O DEJADA @@@
660 IFY<100THEN690
670 CO=1:LINE(26,160)-(39,158),1,BF
:SC=13
680 GOTO 430
690 IFCO<>1THENGOTO430
700 SC=15:PU=PU+1:CO=0:LINE(26,160)
-(39,158),13,BF:LINE(225,166-4*PU)-
(238,166-4*PU+3),13,BF:LINE(230,180
)-(200,172),4,BF:PRINT#1,USING"##";
PU
710 H=(RND(1)*180+10):V=RND(1)*120+
5:DRAW"c2bm=h;,=v;f2e2f3e2fe4f2g5f2
g3f2g2h3g5h3g2h3e5h4e2h3e2f3":PAINT
(H,V+1),2:PAINT(H+8,V+8),2
720 FU=230:LINE(70,185)-(229,187),1
5,BF
730 GOTO 430
740 '@@@@ EXPLOSION @@@@
750 SOUND0,0:SOUND1,5:SOUND2,0:SOUN
D3,13:SOUND4,255:SOUND5,15:SOUND6,3
0:SOUND7,0:SOUND8,16:SOUND9,16:SOUN
D10,16:SOUND11,0:SOUND12,5:SOUND13,
0
760 FORT=1TO30:NEXT:SOUND12,56:SOUN
D13,0:FORT=1TO60:PUTSPRITEVI,(X,Y),
```




```
2,FO%:PUTSPRITEVI,(X,Y),2,6:NEXT:PU
TSPRITEVI,(0,209)
770 LINE(26,160)-(39,158),13,BF:CO=
0:VI=VI-1:IFVI=-1THEN800
780 FU=230:LINE(70,185)-(229,187),1
5,BF:GOTO400
790 '◎◎◎ FINAL ◎◎◎
```

```
800 FORT=1TO1000:NEXT:SCREEN1:COLOR
 15,4,4
810 LOCATE3,10:PRINT"¿OTRA PARTIDA?
 <S/N>"
820 A$=INKEY$:IFA$="s"ORA$="S"THEN2
50ELSEIFA$="n"ORA$="N"THENSCREEN0EL
SE820
```

TEST DE LISTADO

10 - 58	130 -170	250 -101	370 -200	490 -131	610 -101	730 - 70
20 - 58	140 -110	260 - 58	380 - 58	500 -131	620 - 96	740 - 58
30 - 58	150 - 89	270 - 40	390 - 59	510 -124	630 - 29	750 -166
40 - 58	160 - 58	280 - 99	400 - 95	520 -124	640 - 70	760 - 93
50 - 58	170 - 31	290 -254	410 - 82	530 -158	650 - 58	770 -203
60 - 58	180 - 56	300 - 82	420 - 58	540 -125	660 -227	780 -143
70 - 58	190 -112	310 - 74	430 -231	550 -132	670 -240	790 - 58
80 - 58	210 - 74	320 -154	440 -171	560 -106	680 - 70	800 - 28
90 -243	215 -171	330 -234	450 -132	570 -164	690 - 45	810 -172
100 -187	220 - 29	340 -113	460 -132	580 -154	700 -106	820 -212
110 -164	230 - 16	350 - 9	470 -123	590 - 95	710 - 6	TOTAL:
120 -102	240 -185	360 -102	480 -123	600 -110	720 - 45	8895



CRAZY LABYRINTH

Por Manuel J. Muñoz Martínez

Nos encontramos ante un laberinto delirante, puesto que los obstáculos de la pantalla se van creando mientras nosotros intentamos encontrar la salida. No se puede chocar contra las paredes externas, así como tampoco se puede tocar ningún «cubo» o la cola que se forma tras nosotros. Para cambiar de nivel, debe pasarse por encima de un rombo amarillo que aparece aleatoriamente.

```
10 '*******************************
20 '*                             *
30 '*CRAZY LABYRINTH              *
40 '*                             *
50 '*MANUEL JOSE MUÑOZ MARTINEZ*
60 '*                             *
70 '* PARA MSX-EXTRA              *
80 '* ANDUJAR 1986                *
90 '*******************************
100 L=3:SC=0:P=1:H=5000:N$="JOHNNY"
110 GOSUB 1070
120 GOTO 300
130 '
140 '  pone bloques
150 '
160 A%=INT(RND(-TIME)*29)+1:B%=INT
(RND(1)*21)+1:LOCATE A%,B%:PRINT"p"
:B=B+1
170 IF B=20+P*5 THEN GOSUB 700
180 RETURN
190 '
200 ' calcula choque
210 '
220 C%=VPEEK(BASE(5)+((Y%*32)+X%))
230 GOSUB 160
240 PUT SPRITE 0,(X%*8,Y%*8),15,0
250 IF C%<>32 THEN GOSUB 770
260 RETURN
270 '
280 ' inicializacion
290 '
300 CLS:KEY OFF:WIDTH 32:SCREEN 1,,
0:WIDTH 32:CLS
310 COLOR 15,1,1:CLS
320 FOR N=ASC("a")*8 TO ASC("h")*8+
7:READ A$:VPOKEN,VAL("&H"+A$):NEXT
N
330 DATA 3f,7f,ff,ff,f0,f0,f0,f0,ff
,ff,ff,ff,0,0,0,0,fc,fe,ff,ff,f,f,f
,f,f0,f0,f0,f0,f0,f0,f0,f0,f,f,f,f,
f,f,f,f,f0,f0,f0,f0,ff,ff,7f,3f,0,0
,0,0,ff,ff,ff,ff,f,f,f,f,ff,ff,fe,f
```

PROGRAMAS

```
c
340 VPOKE BASE(6)+12,&H46:VPOKE BAS
E(6)+13,&H46
350 LOCATE 0,0:PRINT"abbbbbbbbbbbb
bbbbbbbbbbbbbbbbc":FOR N=1 TO 22:LO
CATE 0,N:PRINT"d":LOCATE 30,N:PRINT
"e":NEXT:LOCATE 0,22:PRINT"fggggggg
ggggggggggggggggggggh"
360 FOR N=ASC("p")*8 TO ASC("p")*8+
7:READ A$:VPOKE N,VAL("&H"+A$):NEXT
 N
370 DATA 0,0,3c,3c,3c,3c,0,0
380 RESTORE 370
390 FOR N=ASC("z")*8 TO ASC("z")*8+
7:READ A$:VPOKE N,VAL("&h"+A$):NEXT
 N
400 FOR N=0 TO 1:FORS=1 TO 8:READ A
$:S$=S$+CHR$(VAL("&h"+A$)):NEXT S:S
PRITE$(N)=S$:S$="":NEXT N
410 DATA ff,81,81,81,81,81,81,ff
420 DATA 18,3c,7e,ff,ff,7e,3c,18
430 VPOKE BASE(6)+14,&HFD
440 VPOKE BASE(6)+15,&HAC
450 X%=28:Y%=21:LOCATE X%,Y%:PRINT"
z":A%=2:B%=21:LOCATE A%,B%:PRINT"p"
460 '
470 ' control de juego
480 '
490 ON SPRITE GOSUB620
500 SPRITE ON
510 D%=STICK(0)
520 IF D%=2 OR D%=4 OR D%=6 OR D%=8
 THEN D%=1
530 IF D%=1THEN Y%=Y%-1:GOSUB 220:L
OCATE X%,Y%:PRINT"z"
540 IF D%=3THEN X%=X%+1:GOSUB 220:L
OCATE X%,Y%:PRINT"z"
550 IF D%=5THEN Y%=Y%+1:GOSUB 220:L
OCATE X%,Y%:PRINT"z"
560 IF D%=7THEN X%=X%-1:GOSUB 220:L
OCATE X%,Y%:PRINT"z"
570 IF STICK(0)<>0 THEN GOTO 510 EL
SE GOTO 530
580 GOTO 510
590 '
600 ' choque con rombo
610 '
620 PUT SPRITE 1,(0,192),1,1
630 FOR N=1 TO 2:SOUND 9,15:SOUND 1
0,15:FOR I=1 TO 52 STEP 8:SOUND 0,I
:SOUND 1,I:SOUND 2,I:SOUND 3,I:SOUN
D 4,I:FOR S=0 TO 30:NEXT S:NEXT I:S
OUND 10,0:SOUND 9,0:NEXT N
640 P=P+1:SC=SC+1000:BEEP
650 GOSUB 990
660 GOTO 470
670 '
680 '  pone rombo
690 '
700 '
710 Q1=0:Q2=0:Q1=INT(RND(-TIME)*29)
+1:Q2=INT(RND(-TIME)*20)+1:PUT SPRI
TE 1,(Q1*8,Q2*8),10,1:LOCATE Q1,Q2:
PRINT" "
720 SOUND 0,0:SOUND 1,0:SOUND 2,0:S
OUND 3,1:SOUND 4,0:SOUND 5,0:SOUND
6,255:SOUND 7,&B10111000:SOUND 8,16
:SOUND 9,16:SOUND 10,0:SOUND 11,0:S
OUND 12,8:SOUND 13,14
730 RETURN
740 '
750 '  explosion
760 '
770 SOUND 11,50:SOUND 12,20:SOUND 1
3,19:SOUND 7,90:SOUND 9,30:SOUND 10
,30:SOUND 8,0:SOUND 6,30
780 CLS:PUT SPRITE 0,(0,192),1,0:PU
T SPRITE 1,(10,192),1,1
790 L=L-1:IF L=0 THEN GOTO 870
800 SC=SC+(B*40)
810 LOCATE 0,0:PRINT"abbbbbbbbbbbb
bbbbbbbbbbbbbbbbc":FOR N=1 TO 21:LO
CATE 0,N:PRINT"d":LOCATE 30,N:PRINT
"e":NEXT
820 LOCATE 0,22:PRINT"fgggggggggggg
gggggggggggggggggh"
830 LOCATE 8,5:PRINT"VIDAS:";L:LOCA
TE 8,7:PRINT"PUNTOS:";SC:LOCATE 8,9
:PRINT"RECORD:";H:LOCATE 8,11:PRINT
"PANTALLA:";P:LOCATE8,13:PRINT"EL M
EJOR:";N$
840 FOR N=0 TO 1000:NEXT N
850 GOSUB 980
860 GOTO 470
870 '
880 '  FIN DE JUEGO
890 '
900 GOSUB 1020
910 SC=SC+(B*40)
920 IF SC>=H THEN LOCATE 8,9:PRINT"
RECORD:";SC
930 IF SC>=H THEN LOCATE 3,15:INPUT
"¿NOMBRE";N$:N$=LEFT$(N$,8):H=SC
940 LOCATE 3,5:PRINT"
                    "
950 LOCATE 3,10:PRINT"¿QUIERES VOLV
ER A JUGAR?"
960 I$=INKEY$:IF I$="S" OR I$="s" T
```



PROGRAMAS

```
HEN RESTORE:SC=0:P=0:L=3:GOTO120ELS
E IF I$="n" OR I$="N" THEN GOTO 106
0
970 GOTO 960
980 '
990 ' INICIALIZA
1000 '
1010 CLS:X%=28:Y%=21:A%=2:B%=21:LOC
ATE X%,Y%:PRINT"z":LOCATE A%,B%:PRI
NT"p":PUT SPRITE 0,(X%*8,Y%*8),15,0
1020 LOCATE 0,0:PRINT"abbbbbbbbbbbb
bbbbbbbbbbbbbbbbbc":FOR N=1 TO 22:L
OCATE 0,N:PRINT"d":LOCATE 30,N:PRIN
T"e":NEXT
1030 LOCATE 0,22:PRINT"fggggggggggg
ggggggggggggggggggh"
1040 B=0:FOR N=4 TO 14:LOCATE 3,N:P
RINT"                           ":NE
XT
1050 RETURN
1060 COLOR 15,4,4:CLS:KEY ON:SCREEN
 0:END
1070 '
1080 ' PRESENTACION
1090 '
1100 CLS:COLOR 4,1,1:CLS:KEY OFF
1110 Z1=8:Z2=1:LOCATE 7,1:PRINT"■":
LOCATE 23,1:PRINT"■"
1120 A$="MANUEL JOSE MUñOZ MARTINEZ
 ....PRESENTA..........CRAZY LABYRI
NTH...."
1130 A$=SPACE$(15)+A$+SPACE$(15)
1140 FOR A=1 TO LEN(A$)
1150 LOCATE Z1,Z2,0
1160 PRINT MID$(A$,A,15)
1170 FOR B=1 TO 100:NEXT
1180 NEXT A
1190 PLAY"T200ABCDEFG","T200GFEDCBA
"
1200 LOCATE 0,10:PRINT"PARA EMPEZAR
 PULSA UNA TECLA"
1210 IF INKEY$="" THEN 1210
1220 RETURN
```

TEST DE LISTADO

10 - 58	190 - 58	370 -208	550 -114	730 -142	910 -201	1090 - 58
20 - 58	200 - 58	380 - 13	560 -115	740 - 58	920 - 48	1100 -241
30 - 58	210 - 58	390 -182	570 -115	750 - 58	930 -155	1110 - 12
40 - 58	220 -240	400 - 6	580 -150	760 - 58	940 - 61	1120 -185
50 - 58	230 - 59	410 -198	590 - 58	770 -105	950 -221	1130 -169
60 - 58	240 -206	420 -134	600 - 58	780 - 59	960 -156	1140 -228
70 - 58	250 - 90	430 - 58	610 - 58	790 -118	970 - 90	1150 - 88
80 - 58	260 -142	440 -234	620 -145	800 -201	980 - 58	1160 -128
90 - 58	270 - 58	450 - 4	630 - 20	810 - 21	990 - 58	1170 -206
100 -135	280 - 58	460 - 58	640 -217	820 -194	1000 - 58	1180 -196
110 -205	290 - 58	470 - 58	650 -124	830 -241	1010 -229	1190 -249
120 -196	300 -181	480 - 58	660 -110	840 -187	1020 - 22	1200 - 71
130 - 58	310 - 48	490 -101	670 - 58	850 -114	1030 -194	1210 - 80
140 - 58	320 -107	500 - 92	680 - 58	860 -110	1040 -178	1220 -142
150 - 58	330 -236	510 - 91	690 - 58	870 - 58	1050 -142	
160 -132	340 -129	520 -108	700 - 58	880 - 58	1060 -156	
170 - 92	350 - 18	530 -111	710 - 6	890 - 58	1070 - 58	TOTAL:
180 -142	360 -130	540 -110	720 - 74	900 -154	1080 - 58	13142

msxclub

DE PROGRAMAS

ESPECIAL SOFTWARE



Prepárate pues estamos trabajando en un número especial con todo el SOFTWARE MSX del mercado español. Y también noticias, novedades, comentarios y un largo etcétera que harán de este MSX CLUB ESPECIAL SOFTWARE algo FUERA DE SERIE.

CALCULADORA

Por Javier Esquirol

Este práctico programa convierte a tu MSX en una calculadora científica que va a realizar un montón de operaciones que podrás seleccionar sencillamente colocando el cursor sobre la cifra u operación que desees efectuar y pulsar return.

```
1 REM*****************
2 REM** CALCULADORA **
3 REM**    PARA     **
4 REM**  MSX-EXTRA  **
5 REM**    POR      **
6 REM*JAVIER ESQUIROL*
7 REM*****************
10 SCREEN 1,1:WIDTH 32:CLEAR
20 KEY OFF:COLOR 15,1,1:CLS
30 DIM NI$(30,23),PI(30,23)
40 REM''''''Llenar Nombre Item $'''''
''
50 NI$(23,13)="7"
60 NI$(25,13)="8"
70 NI$(27,13)="9"
80 NI$(23,15)="4"
90 NI$(25,15)="5"
100 NI$(27,15)="6"
110 NI$(23,17)="1"
120 NI$(25,17)="2"
130 NI$(27,17)="3"
140 NI$(23,19)="Ø"
150 NI$(25,19)="."
160 NI$(27,11)="E"
170 REM ''''''Llenar Pos. Item'''''
''
180 PI(4,7)=4:PI(8,7)=4:PI(12,7)=4:
PI(16,7)=4
190 PI(4,9)=4:PI(8,9)=4:PI(12,9)=4:
PI(16,9)=4
200 PI(4,11)=4:PI(8,11)=4:PI(12,11)
=4:PI(16,11)=4
210 PI(4,13)=4:PI(8,13)=4:PI(12,13)
=4:PI(16,13)=4
220 PI(4,15)=3:PI(8,15)=3:PI(12,15)
=3:PI(16,15)=3
230 PI(4,19)=4:PI(8,19)=4:PI(12,19)
=4:PI(16,19)=4
240 PI(22,6)=4:PI(24,6)=4:PI(26,6)=
4:PI(28,6)=4
250 PI(22,8)=4:PI(24,8)=4:PI(26,8)=
4:PI(28,8)=4
260 PI(4,17)=4:PI(8,17)=4:PI(12,17)
=4:PI(16,17)=4
270 PI(29,11)=3:PI(29,13)=3:PI(29,1
5)=3:PI(29,17)=3:PI(29,19)=3
280 PI(23,11)=4:PI(25,11)=4:PI(27,1
1)=2
290 PI(23,13)=2:PI(25,13)=2:PI(27,1
3)=2
300 PI(23,15)=2:PI(25,15)=2:PI(27,1
5)=2
310 PI(23,17)=2:PI(25,17)=2:PI(27,1
7)=2
320 PI(23,19)=2:PI(25,19)=2
330 PI(27,19)=1
340 REM''''''Diseño Calculadora'''''
''
350 FOR X=1 TO 30:LOCATEX,1:PRINT"▔
";:NEXT X
360 FOR X=1 TO 31:LOCATEX,21:PRINT"
▀";:NEXT X
370 LOCATE1,21:PRINT" ▀":LOCATE31,21
:PRINT"▀ "
380 FOR Y=1 TO 20:LOCATE1,Y:PRINT" ▌
";:NEXT Y
390 FOR Y=1 TO 20:LOCATE31,Y:PRINT"
▌ ";:NEXT Y
400 FOR X=2 TO 24:LOCATEX,2:PRINT"▔
";:NEXT X
410 FOR X=2 TO 24:LOCATEX,4:PRINT"▁
";:NEXT X
420 FOR Y=2 TO 4:LOCATE2,Y:PRINT" ▌"
;:NEXT Y
430 FOR Y=2 TO 4:LOCATE24,Y:PRINT"▌
";:NEXT Y
440 LOCATE 25,2:PRINT"CALCC"
450 LOCATE 25,4:PRINT"XT-Ø1"
460 LOCATE 23,11:PRINT"i S E /"
470 LOCATE 23,13:PRINT"7 8 9 *"
480 LOCATE 23,15:PRINT"4 5 6 -"
490 LOCATE 23,17:PRINT"1 2 3 +"
500 LOCATE 23,19:PRINT"Ø . C ="
510 LOCATE 21,9:PRINT" r"
520 FOR Y=1Ø TO 2Ø:LOCATE21,Y:PRINT
```

PROGRAMAS

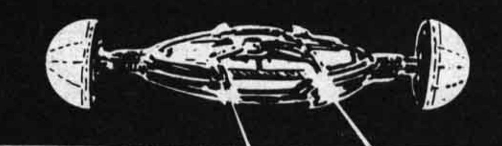
```
"|";:NEXT Y
530 FOR X=22 TO 30:LOCATEX,9:PRINT"
W";:NEXT X
540 FOR X=3 TO 19:LOCATEX,6:PRINT"-
";:NEXT X
550 LOCATE20,6:PRINT"┐":LOCATE2,6:P
RINT"┌"
560 FOR Y=7 TO 19:LOCATE20,Y:PRINT"
V";:NEXT Y
570 FOR Y=7 TO 19:LOCATE2,Y:PRINT"|
";:NEXT Y
580 FOR X=3 TO 19:LOCATEX,20:PRINT"
W";:NEXT X
590 FOR X=3 TO 19:LOCATEX,18:PRINT"
W";:NEXT X
600 LOCATE2,20:PRINT"└":LOCATE20,20
:PRINT"┘":LOCATE2,18:PRINT"├":LOCAT
E20,18:PRINT"┤"
610 LOCATE 4,7:PRINT"SIN COS TAN SQ
R"
620 LOCATE 4,9:PRINT"ASN ACS ATN X^
2"
630 LOCATE 4,11:PRINT"SNH CSH TNH F
AC"
640 LOCATE 4,13:PRINT"LOG LN  EXP 1
0^"
650 LOCATE 4,15:PRINT"X%Y X\Y X√Y X
^Y"
660 LOCATE 4,17:PRINT"ABS INT FIX B
SP"
670 LOCATE 4,19:PRINT"DEG RAD GRA π
"
680 LOCATE 22,6:PRINT"R R R SM"
690 LOCATE 22,8:PRINT"M M M CM"
700 REM '''''''Diseño Sprite'''''''
''
710 GOSUB 1100
720 REM ''''Control De Operaciones'
''
730 I=22:J=10:Z=3:FC=3.141592654#/1
80
740 PUT SPRITE2,(I*8,J*8),6,0
750 IF STICK(0)=1 THEN J=J-1
760 IF STICK(0)=2 THEN J=J-1:I=I+1
770 IF STICK(0)=3 THEN I=I+1
780 IF STICK(0)=4 THEN I=I+1:J=J+1
790 IF STICK(0)=5 THEN J=J+1
800 IF STICK(0)=6 THEN J=J+1:I=I-1
810 IF STICK(0)=7 THEN I=I-1
820 IF STICK(0)=8 THEN I=I-1:J=J-1
830 I=IMOD31:J=JMOD31
840 IF I<0 THEN I=31+I
850 IF J<0 THEN J=31+J
860 IF STRIG(0) THEN 870 ELSE 740
870 IF PI(I,J)=0 THEN 740
880 IF PI(I,J)=1 THEN LOCATE3,3:PRI
NTSPC(20):Z=3:B$="":A$="":A=0:GOTO
740
890 FOR W=0 TO 25:NEXT W
900 IF I=29 AND J=19 THEN 1050
910 IF I=16 AND J=17 AND A$<>"" THE
N A$=MID$(A$,1,LEN(A$)-1):A=VAL(A$)
:LOCATE3,3:PRINTA:Z=Z-1:GOTO 740
920 IF I=29 AND J=15 THEN 930 ELSE
940
930 IF MID$(A$,LEN(A$),1)="E" THEN
A$=A$+"-":LOCATEZ+1,3:PRINT"-":Z=Z+
1:GOTO 740
940 IF PI(I,J)=2 THEN 950 ELSE 990
950 IF CO=1 THEN A=0:A$="":LOCATE3,
3:PRINTSPC(20):Z=3:CO=0
960 IF LEN(A$)>18 THEN 740
970 A$=A$+NI$(I,J)
980 Z=Z+1:LOCATEZ,3:PRINTNI$(I,J);:
GOTO 740
990 IF PI(I,J)=4 THEN 1000 ELSE 102
0
1000 A=VAL(A$):GOSUB 1210:CO=1:IF E
R=1 THEN A=0:A$="":LOCATE3,3:PRINT"
 Error":Z=3:ER=0:GOTO 740
1010 LOCATE3,3:PRINTA:A$=STR$(A):Z=
3:GOTO 740
1020 IF B$="" THEN 1040
1030 A=VAL(A$):B=VAL(B$):GOSUB1730:
B$=STR$(A):U=I:V=J:LOCATE3,3:PRINTA
:Z=3:CO=1:GOTO 740
1040 A=VAL(A$):LOCATE3,3:PRINTSPC(2
0):FOR W=1 TO 50:NEXT:LOCATE3,3:PRI
NTA:U=I:V=J:Z=3:B$=A$:A$="":CO=1:GO
TO 740
1050 A=VAL(A$):B=VAL(B$):GOSUB 1730
1060 IF ER=1 THEN A=0:A$="":LOCATE3
,3:PRINT" Error":Z=3:ER=0:GOTO 740
1070 LOCATE3,3:PRINTA:B$="":A$=STR$
(A):Z=3:CO=1:GOTO 740
1080 GOTO 740
1090 END
1100 REM''''Sub. Diseño Sprite'''''
'
1110 P$=""
1120 FOR I=0 TO 2
1130 READ D$
1140 P$=P$+CHR$(VAL("&B"+D$))
1150 NEXT I
1160 SPRITE$(0)=P$
1170 RETURN
```




```
1180 DATA 1110000Ø
1190 DATA 1110000Ø
1200 DATA 1110000Ø
1210 REM ''''Sub. Oper. Tipo 4'''''
''
1220 LOCATE4,3:PRINTSPC(2Ø)
1230 IF J<>7 THEN 128Ø
1240 IF I=4 THEN A=SIN(A*FC)
1250 IF I=8 THEN A=COS(A*FC)
1260 IF I=12 THEN A=TAN(A*FC)
1270 IF I=16 THEN ON ERROR GOTO 172
Ø:A=SQR(A)
1280 IF J<>9 THEN 133Ø
1290 IF I=4 THEN ON ERROR GOTO 172Ø
:A=ATN(A/SQR(1-A^2)):A=A/FC
1300 IF I=8 THEN ON ERROR GOTO 172Ø
:A=-ATN(A/SQR(1-A^2))+1.57Ø8:A=A/FC
1310 IF I=12 THEN ON ERROR GOTO 172
Ø:A=ATN(A):A=A/FC
1320 IF I=16 THEN ON ERROR GOTO 172
Ø:A=A^2
1330 IF J<>11 THEN 146Ø
1340 IF I=4 THEN ON ERROR GOTO172Ø:
X=X*FC:A=(EXP(X)-EXP(-X))/2
1350 IF I=8 THEN ON ERROR GOTO172Ø:
X=X*FC:A=(EXP(X)+EXP(-X))/2
1360 IF I=12 THEN ON ERROR GOTO172Ø
:X=X*FC:A=(EXP(X)-EXP(-X))/(EXP(X)+
EXP(-X))
1370 IF I=16 THEN 138Ø ELSE 144Ø
1380 IF A=Ø OR A=1 THEN A=1:GOTO 14
4Ø
1390 IF A>48 THEN ER=1:GOTO 144Ø
1400 IF A-INT(A)<>Ø THEN 144Ø
1410 FOR TG=A-1 TO 1 STEP-1
1420 ON ERROR GOTO172Ø:A=A*TG
1430 NEXT TG
1440 IF I=23 THEN ON ERROR GOTO172Ø
:A=1/A
1450 IF I=25 THEN A=A*-1
1460 IF J<>13 THEN 151Ø
1470 IF I=4 THEN ON ERROR GOTO 172Ø
:A=LOG(A)/LOG(1Ø)
1480 IF I=8 THEN ON ERROR GOTO 172Ø
:A=LOG(A)
1490 IF I=12 THEN ON ERROR GOTO 172
Ø:A=2.718281828#^A
1500 IF I=16 THEN ON ERROR GOTO 172
Ø:A=1Ø^A
1510 IF J<>17 THEN 155Ø
1520 IF I=4 THEN A=ABS(A)
1530 IF I=8 THEN A=INT(A)
1540 IF I=12 THEN A=FIX(A)
1550 IF J<19 THEN 16ØØ
1560 IF I=4 THEN FC=3.141592654#/18
Ø
1570 IF I=8 THEN FC=1
1580 IF I=12 THEN FC=3.141592654#/2
ØØ
1590 IF I=16 THEN A=3.1415926535898
#
1600 IF J<>3 THEN 161Ø
1610 IF J<>6 THEN 166Ø
1620 IF I=22 THEN A=M1
1630 IF I=24 THEN A=M2
1640 IF I=26 THEN A=M3
1650 IF I=28 THEN M1=M1+A
1660 IF J<>8 THEN 171Ø
1670 IF I=22 THEN M1=A
1680 IF I=24 THEN M2=A
1690 IF I=26 THEN M3=A
1700 IF I=28 THEN M1=Ø:M2=Ø:M3=Ø
1710 RETURN
1720 IF ERR THEN ER=1:RESUME NEXT:R
ETURN
1730 REM ''''''Sub. Oper. Tipo 3'''
''
1740 LOCATE3,3:PRINTSPC(2Ø)
1750 IF U<>29 THEN 18ØØ
1760 IF V=11 THEN ON ERROR GOTO 186
Ø:A=B/A
1770 IF V=13 THEN ON ERROR GOTO 186
Ø:A=B*A
1780 IF V=15 THEN ON ERROR GOTO 186
Ø:A=B-A
1790 IF V=17 THEN ON ERROR GOTO 186
Ø:A=B+A
1800 IF V<>15 THEN 185Ø
1810 IF U=4 THEN ON ERROR GOTO 186Ø
:A=A*B/1ØØ
1820 IF U=8 THEN ON ERROR GOTO 186Ø
:A=A*B/1ØØØ
1830 IF U=12 THEN ON ERROR GOTO 186
Ø:A=A^(1/B)
1840 IF U=16 THEN A=B^A
1850 RETURN
1860 IF ERR THEN ER=1:RESUME NEXT:R
ETURN
```

TEST DE LISTADO

1	Ø	1Ø	-2Ø2	8Ø	-227	15Ø	-227
2	Ø	2Ø	- 33	9Ø	-23Ø	16Ø	-244
3	Ø	3Ø	-166	1ØØ	-233	17Ø	- Ø
4	Ø	4Ø	- Ø	11Ø	-226	18Ø	-222
5	Ø	5Ø	-228	12Ø	-229	19Ø	-23Ø
6	Ø	6Ø	-231	13Ø	-232	2ØØ	-23Ø
7	Ø	7Ø	-234	14Ø	-227	21Ø	-238



PROGRAMAS

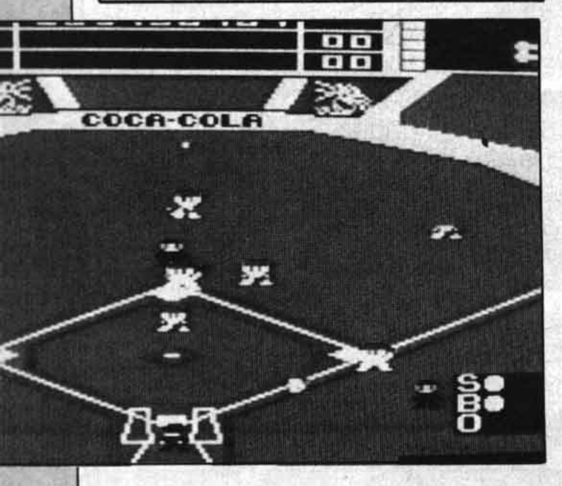
22Ø -242	46Ø -131	69Ø -2ØØ	93Ø - 99	117Ø -142	141Ø - 28	165Ø -229
23Ø - 6	47Ø - 39	7ØØ - Ø	94Ø -161	118Ø - 7	142Ø -2Ø1	166Ø -1Ø4
24Ø - 18	48Ø - 35	71Ø -235	95Ø -2Ø2	119Ø - 7	143Ø - 3Ø	167Ø -112
25Ø - 26	49Ø - 26	72Ø - Ø	96Ø -175	12ØØ - 7	144Ø - 4	168Ø -115
26Ø -254	5ØØ - 57	73Ø -215	97Ø -117	121Ø - Ø	145Ø - 45	169Ø -118
27Ø -215	51Ø -17Ø	74Ø - 97	98Ø -218	122Ø - 35	146Ø -162	17ØØ -185
28Ø -134	52Ø - 81	75Ø -24Ø	99Ø -243	123Ø -179	147Ø -174	171Ø -142
29Ø -136	53Ø - 91	76Ø -175	1ØØØ -193	124Ø -12Ø	148Ø -2Ø3	172Ø - 11
3ØØ -142	54Ø - 6Ø	77Ø -239	1Ø1Ø - 56	125Ø -127	149Ø -185	173Ø - Ø
31Ø -148	55Ø -119	78Ø -176	1Ø2Ø - 32	126Ø -13Ø	15ØØ - 5	174Ø - 34
32Ø -252	56Ø - 78	79Ø -243	1Ø3Ø -155	127Ø -2Ø6	151Ø -2Ø7	175Ø -225
33Ø - 99	57Ø - 82	8ØØ -179	1Ø4Ø -163	128Ø -236	152Ø -249	176Ø -194
34Ø - Ø	58Ø - 72	81Ø -244	1Ø5Ø -2Ø3	129Ø - 11	153Ø -252	177Ø -195
35Ø -171	59Ø - 7Ø	82Ø -182	1Ø6Ø -174	13ØØ - 85	154Ø - 26	178Ø -196
36Ø -218	6ØØ - 78	83Ø -144	1Ø7Ø -216	131Ø -249	155Ø - 21	179Ø -197
37Ø -144	61Ø -232	84Ø - 79	1Ø8Ø -125	132Ø -255	156Ø -2Ø7	18ØØ - 6
38Ø -191	62Ø -198	85Ø - 82	1Ø9Ø -129	133Ø -11Ø	157Ø - 64	181Ø - 34
39Ø -218	63Ø -189	86Ø -175	11ØØ - Ø	134Ø -163	158Ø -213	182Ø -186
4ØØ -167	64Ø -1Ø8	87Ø - 2	111Ø -167	135Ø -166	159Ø -128	183Ø - 26
41Ø -166	65Ø - 69	88Ø -159	112Ø -183	136Ø - 48	16ØØ -255	184Ø -112
42Ø -179	66Ø -213	89Ø -238	113Ø -239	137Ø -193	161Ø - 52	185Ø -142
43Ø -198	67Ø -174	9ØØ - 7Ø	114Ø -151	138Ø -151	162Ø -112	186Ø - 11
44Ø -164	68Ø -229	91Ø -134	115Ø -2Ø4	139Ø -225	163Ø -115	TOTAL:
45Ø -138		92Ø - 97	116Ø -176	14ØØ - 8Ø	164Ø -118	26244

BIT-BIT Software Juegos

POR CLAUDIA TELLO HELBLING

BASEBALL CRAZE

Hudson Soft/Serma
Formato: Tarjeta
Mandos: Teclado



Este es un juego que ha de hacer las delicias de todos aquellos aficionados a los deportes en general y sobre todo para los acérrimos amantes del beisbol ya que reproduce con pasmosa exactitud las reglas de dicho deporte. Su realismo es extraordinario de modo que permite aplicar las estrategias de defensa y ataque de un juego que junto al «football americano» es considerado rey en los EE.UU. y deporte casi nacional en Japón. El jugador dispone de dos opciones, jugar contra la máquina, alternándose en la defensa y ataque del campo, o bien jugar contra otro compañero. Otra de las características que lo hacen definitivamente apasionante es que cada jugador puede organizar su equipo conforme a las particulares peculiaridades de los jugadores, así el jugador se convierte en un seleccionador que puede llegar a conseguir que su equipo tenga unas características, alternando el orden de bateo. Por otra parte el pitcher —o lanzador—, es también programable con lo que puede conseguir efectos de curvas, o «softballs», según desee. Los bateadores, a elección, podrán ser diestros o zurdos con lo que su similitud con el juego real tal como he dicho es asombrosa. Por último, el juego consta de dos pantallas alternativas, una que reproduce en primer plano lo que sucede en el «diamante» y otra que da una visión general del campo cuando la pelota bateada alcanza lo que en el beisbol se denomina hit. Por ello **Baseball Craze** merece una alta conceptuación como juego divertido a la par que difícil.

Sonido: Adecuado
Grafismo: Muy real
Conclusiones: Juego de gran realismo muy entretenido.
Precio aproximado: 8.500 pts.– incluyendo cartucho adaptador.

FLOPPY

Manhattan Transfer, S.A.
Formato: Cassette 32K
Mandos: Teclado
Carga: LOAD "CAS:",R

FLOPPY, el preguntón es mucho más que un juego. Es un divertimento educativo que te permite poner a prueba tus conocimientos de historia y geografía. La cosa comienza cuando sobre el cielo de España aparece un avión del cual se lanza en paracaídas nuestro amigo Floppy. Pero como Floppy no sabe ni jota de historia ni de geografía españolas empieza a preguntarte de una cosa u otra alternativamente. Floppy se desplaza a cualquier punto del territorio de la península o de las islas y te pregunta esto y aquello, pero su paciencia tiene un límite de modo que si no le contestas la consulta tú pierdes puntos, lo mismo que si tú le preguntas al ordenador cuál es la respuesta acertada. Pero a Floppy no le hace mucha gracia que tú pierdas, pues él sólo tiene tres vidas de modo que tienes que aguzar mucho el ingenio y refrescar bastante la memoria.



Realmente las preguntas no son fáciles en su mayoría y de nada te sirve haberlas respondido en otras ocasiones, pues aparecen de modo aleatorio en una ventana que está a la izquierda de tu pantalla. Por esto te recomiendo que prestes mucha atención y aprende que es allí donde ganarás los verdaderos puntos que harán vivir más tiempo a **Floppy, el preguntón.**

Gráficos: Apropiados
Conclusiones: Un juego entretenido y muy útil para los que van a EGB.
Precio: 1.000 pts.

CHILLER

Mastertronic/Drosot
Formato: cassette 32K
Mandos: Joystick o teclado
Carga: LOAD "CAS:",R

Este es un juego en el que puedo destacar tanto la calidad de su grafismo como las dificultades de alcanzar el objetivo. Este consiste en rescatar a tu chica que ha sido raptada y escondida en un lugar más allá del bosque de los malos espíritus. Claro que si este fuese un juego como los demás, la cosa terminaría allí, pero no es este el caso, pues una vez rescatada la dama en cuestión hay que hacer el camino inverso salvando todos los obstáculos y recogiendo las mismas cruces necesarias para abrir los caminos. Además una vez que tienes a la chica la superación de las pruebas debe hacerse con ambos alternativamente. El juego comienza cuando tu coche se queda sin gasolina —tal vez por la influencia de los malos espíritus—, a la orilla misma del bosque encantado. Pero no tienes más alternativa que seguir adelante y tratar de llegar a la mansión donde está tu chica. Para ello tienes que seguir adelante y recoger todas las cruces evitando que te toquen los murciélagos, cadáveres resucitados y otros espectros pues te irán quitando la energía, lo mismo que el andar y saltar. Para reponerla tienes que comerte las setas que veas tanto en el suelo como en los árboles, pero ¡ojo! algunas son venenosas. Una vez que has recogido todas las cruces mágicas saldrás del bosque.

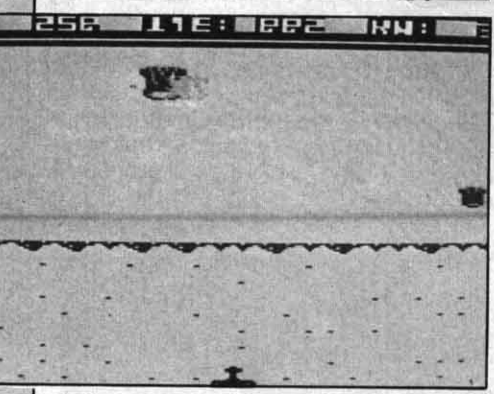
La segunda pantalla es el cine a la orilla del bosque donde te acechan más espectros; la tercera el ghetto; la cuarta el cementerio y la quinta la mansión donde está tu amada. Pero tu misión no ha terminado pues ahora te toca desandar todo el peligroso camino y además cuidando de tu chica. Claro que ella no permanecerá im-



pasible ante los peligros y alternativamente te ayudará a superarlos. Cuando le toque el turno a uno u a otro lo sabrás por el color del borde. La chica debe recoger las cruces mágicas rojas y el chico las azules.

Sonido: Insistente
Grafismo: Muy bueno
Conclusiones: Un juego que requiere mucha habilidad y sangre fría para superar los numerosos obstáculos que se te presentan.
Precio aproximado: 750 pts.

MAD FOX



Manhattan Transfer, S.A.
Formato: Cassette 32K
Mandos: Joystick o teclado
Carga: CLOAD "CAS:"

Este juego diseñado por Rubén Jiménez, autor también de Krypton y Star Runner entre otros, es para poner a prueba los reflejos y la habilidad como conductores. Ambientado en la Era Post Atómica se trata de una travesía por vasto desierto cubierto de cráteres. Diseminados por él hay bidones de gasolina, imprescindibles para la subsistencia de los pocos terrestres que aún quedan después del desastre nuclear. La tarea de Mad Fox es recoger esos bidones, pero estos no siempre son fáciles de recoger ya que tiene que imprimir a su vehículo una alta velocidad, pues también está en su contra el tiempo. Y por si esto fuera poco, después de algunos kilómetros aparece una avioneta pilotada por un maldito egoísta que quiere destruirte lanzándote granadas. Pero no queda ahí la cosa, ya que tras un recorrido importante (no te digo los kilómetros pues es un secreto), aparece una grieta enorme que deberás saltar con tu coche. Para lograrlo sólo tienes que pulsar el botón de fuego del joystick o bien la barra espaciadora.

Toda la información que necesitas durante tu recorrido la tienes en la parte inferior de la pantalla, es decir, el tiempo, el nivel de combustible, la gasolina que recoges y la que consumes, y también los kilómetros recorridos.

Mad Fox presenta diez niveles de dificultad. En principio tú puedes elegir entre amateur y profesional y en cada uno de ellos tienes cinco grados de dificultad. A modo de advertencia te digo que el nivel más bajo, es decir el 1 de amateur se las trae en grande así que ya te puedes imaginar los otros.



Sonido: Apropiado
Gráficos: Correctos
Conclusiones: Sumamente entretenido y de gran impacto por sus efectos de scroll tridimensional.
Precio: 1.000 pts.

STAR SEEKER

Mirrorsoft/Discovery
Formato: Cassette 32K
Mandos: Teclado
Carga: Cara A LOAD "CAS:STAR",R y CARA B LOAD "CAS:SS1",R



Este es un excelente programa de introducción a la astronomía que contiene dos posibilidades de estudio e investigación. Diseñado con un alto sentido didáctico y una gran pulcritud gráfica STAR SEEKER nos da la posición de más de 500 estrellas, muestra mapas estelares vistos desde cualquier punto de la Tierra y en cualquier momento del siglo. El programa representa además constelaciones, muestra el cambio de posición relativa de las estrellas según la rotación de nuestro planeta, presenta los datos referentes a la posición de las distintas estrellas. Con él se puede obtener una ampliación de determinado fragmento del cielo u observar la entrada o salida de las estrellas, trazar planos sobre las observaciones celestes.

El programa de la Cara B llamado Solar System te permite la localización de los planetas en el cielo y dibuja sus órbitas alrededor del sol. Proporciona además toda la información necesaria sobre el sol y la luna y describe la órbita del cometa Halley.

Toda la información, textual o gráfica, que nos da **Star Seeker** el estudiante puede volcarla en impresora, de modo que siempre puede tenerla a mano. A modo de complemento se adjunta un pequeño manual de uso, que contiene asimismo información complementaria sobre varios conceptos astronómicos, estrellas, planetas y otros cuerpos celestes. Su acceso es muy fácil.



Gráficos: Muy cuidados
Conclusiones: Excelente programa de introducción astronómica. Fácil empleo.
Precio aproximado: 2.900 pts.

INICIACION AL LENGUAJE MAQUINA

DEL HARD AL SOFT 16

ROTACION Y DESPLAZAMIENTO

OPERACIONES EN BLOQUE

Normalmente actúan sobre más de uno o dos bytes, como las anteriores. (Ver número anterior).

–CPI (ComPare with Increment: comparación con incremento).

Es la más sencilla de estas instrucciones aunque sólo actúa sobre un byte. Compara el contenido del registro A con el (HL), y automáticamente incrementa el par HL. Así, después de ser ejecutada la instrucción, el par HL señalará el byte siguiente para efectuar otra comparación.

Normalmente el CPI sirve para buscar en la memoria un determinado byte, una vez encontrado se activa el flag Z.

–CPD: La instrucción CPD realiza una función parecida a CPI, pero ésta disminuye el registro HL en lugar de aumentarlo.

Tanto CPI como CPD decrementan el par de registros BC, cosa muy útil, ya que nos permite buscar los bytes en zonas de memoria de determinada longitud.

El principio de la zona en que buscaremos dichos bytes está en el registro HL, y el número de bytes a examinar está en el registro BC. Veamos un ejemplo:

```
        LD    A, (&HF7F8)
        LD    BC,255
        LD    HL, 0000
BUSCA   CPI
        JR    Z, SALIDA
        INC   C
        DEC   C
        JR    NZ, BUSCA
SALIDA  LD    (&HF7F8), HL
        LD    A,2
        LD    (&HF663), A
        RET
```

Estamos examinando el registro menor del par BC, o sea, C. Hemos escrito la instrucción INC seguida de DEC lo que nos permite saber si es 0 el registro C. DEC activa el flag Z y así sabemos si en C hay un 0.

Este programa devuelve el valor 255 si no ha encontrado el byte buscado en los 255 primeros bytes de la memoria, pero si lo encuentra, el programa nos devuelve la posición de dicho byte.

Instrucciones CPIR y CPDR:

Ocupan dos bytes cada una, equivale a CPI y CPD pero con un salto.

La UCP busca automáticamente en una zona de memoria hasta que encuentra el byte o hasta el fin de dicha zona. El registro A especifica el byte que se va a buscar, el par HL tiene el principio de la zona, y el registro BC el número de bytes que ha de examinar.

Así la instrucción puede acabar por:

a) Porque encuentre el byte buscado

b) Porque se llegue al final de la zona

Para saber qué ha causado el fin de la ejecución veremos:

1) Al terminar la instrucción anterior el resultado del registro BC es 0 con lo que se ejecutará un salto a la etiqueta «END», lo que nos indica que se ha examinado todo el bloque sin encontrar el byte buscado.

2) Si se ha encontrado el byte la instrucción se parará.

(Para que te hagas una idea de lo que tarda en ejecutarse esta instrucción diremos que el último programa tardó 0.12 segundos para buscar el byte 101 entre 12.990; más rápido que usando rutinas escritas con instrucciones separadas: CP y JR).

Movimientos de bloques

Se trata de mover fragmentos de memoria en el ordenador.

El lenguaje máquina Z-80 tiene instrucciones que hacen transferencias de bytes directamente.

La primera instrucción que vamos a ver es LDI:

```
        LD    HL, 0000
        LD    DE, 58000
        LD    BC, 100
BUCLE   LDI
        LD    A,B
        OR    C
        JR    NZ, BUCLE
        RET
```

El programa transfiere los 100 bytes que hay desde el 0 en adelante, a la posición 58000 a 58100. El registro DE se denomina normalmente DESTINO (tiene la primera dirección donde van a transferirse los bytes) y al registro HL se le llama FUENTE (tiene la dirección del primer byte que se quiere copiar).

Una vez ejecutada la instrucción LDI los registros HL y DE se incrementan y BC decrementará. La instrucción LDD provoca algo parecido, pero con ella los registros HL y DE tienden a disminuir.

Así, después de ejecutarse la instrucción LDI:

HL: indicará posición 0001

DE: indicará posición 58001

Si después de ejecutar LDI y LDD el par de registros BC no es igual a 0, se activará el flag P/V, y usando dicho flag sabremos si hemos de repetir la instrucción o no.

Existen, no obstante, un método más eficiente para repetir las instrucciones LDI o LDD, se trata de las instrucciones LDIR o LDDR, que examinan automáticamente el flag, repitiendo el proceso si es necesario:



Por ejemplo:

```
LD      HI, 0000
LD      BC, 1000
LD      DE, 58000
LDIR
RET
```

Transmite automáticamente 1000 bytes desde 0 a 1000 a la dirección 58000 en adelante.

Hemos terminado así las instrucciones que operan con bucles, saltos y zonas de memorias o bloques.

Observa ahora una tabla que sintetiza algunas de las características de las instrucciones que hemos visto:

J. C. González Santamaría

EFECTOS EN LOS FLAGS

Mnemónico	Bytes	Tiempo	C	Z	P/V	S	N	H
LDI	2	16	-	-	*	-	0	0
LDD	2	16	-	-	*	-	0	0
LDIR	2	21/16	-	-	0	-	0	0
LDDR	2	21/16	-	-	0	-	0	0
CPI	2	16	-	*	*	*	1	*
CPD	2	16	-	*	*	*	1	*
CPIR	2	21/16	-	*	*	*	1	*
CPDR	2	21/16	-	*	*	*	1	*

*... indica que una operación ha afectado el flag
0... indica que el flag está a 0
1... indica que el flag está a 1
-... indica que no ha sido afectado ese flag

SONIMAG 86'



Coincidiendo con la aparición de este número de tu revista en la calle tendrá lugar en Barcelona la vigesimocuarta edición del Salón Internacional de la Imagen, el Sonido y la Electrónica (SONIMAG), que se prolongará hasta el próximo día 21 de septiembre.

Como cada año, participarán los principales fabricantes de electrónica de consumo, así como los de todos aquellos productos relacionados con ella de una manera u otra.

Es ya tradición que cada nueva edición de esta Feria supere a las anteriores en la presentación de tecnologías punta, y se espera una nutrida asistencia de representantes de todos los puntos del globo.

Ah, naturalmente MANHATTAN TRANSFER, S.A. dispondrá de un Stand en el recinto por donde esperamos ver a todos los amigos que contribuyen mes a mes a que nuestra revista sea líder indiscutible en el terreno del MSX.

Va a haber sorpresas (como viene siendo habitual en SONIMAG).

Si estás en Barcelona del 15 al 21 de septiembre, no dudes en visitar esta Feria. No lo lamentarás en absoluto.

PRIMERAS JORNADAS DEL MSX

Coincidiendo con la celebración del SONIMAG en Barcelona, y en su marco, tendrán lugar los días 15 a 21 de septiembre las PRIMERAS JORNADAS DEL MSX organizadas por MANHATTAN TRANSFER, S.A.

Con la celebración de estas primeras jornadas se pretende agrupar a todos aquellos profesionales relacionados de un modo u otro con el standard MSX.

Como objetivo prioritario de estas Primeras Jornadas figura la unificación de políticas comerciales y de distribución tanto de Hardware como de software. Para ello se presentarán diversas ponencias y se crearán varias comisiones encargadas de agilizar la comunicación entre los diferentes sectores que integrarán las mismas.

Se cuenta con la asistencia de los representantes de las primeras marcas de todos los sectores relacionados con la norma, así como con la participación de la prensa especializada. Hasta la fecha, no se había realizado ningún otro intento de relacionar a los diferentes sectores que componen el espectro del MSX en España. Por ello, y para intentar una mayor cobertura de las necesidades del público, que es en último extremo a quien se pretende beneficiar, tenemos mucha confianza depositada en esta iniciativa.

Esperamos que de estas Jornadas surjan actuaciones conjuntas de todos los integrantes de la norma, y que sus resultados tengan un eco inmediato entre el público.

DOS NUEVOS LADRILLOS PARA LA TORRE DE BABEL

Hace pocos días se ha recibido en esta redacción una nota de prensa de la firma AMSTRAD ESPAÑA, S.A. en la que se comunican las novedades que van a presentar en la próxima edición de SONIMAG, y todavía no hemos salido de nuestro asombro.

La primera «novedad» es que en este SONIMAG se va a presentar el SINCLAIR SPECTRUM + 2, **diseñado básicamente como ordenador de videojuegos e introducción a la informática personal.** De esto cabe deducir un par de cosas:

En primer lugar, el hecho de hablar del nuevo diseño de este SPECTRUM nos hace suponer que asistiremos a la presentación de un nuevo «incompatible» de SINCLAIR. En segundo lugar, viene a confirmar el pensamiento que rondaba nuestras mentes (y las de muchos): El SPECTRUM sólo sirve para jugar, (esperemos que mejor que sus antecesores, no obstante).

La segunda novedad es la presentación del AMSTRAD PC compatible con diversas configuraciones. No sabemos a qué configuraciones se refiere, pero sí sabemos que los diferentes modelos de esta marca que están en el mercado tienen serios problemas de compatibilidad entre sí. Esperemos que se trate del «comodín universal» que solucione todas las incompatibilidades entre sus modelos.

Como puede verse, la novedad es relativa pues sólo se refiere a la aparición de dos «incompatibles». Desgraciadamente, la incompatibilidad ya no es ninguna novedad.

ESPECIAL SOFTWARE

En esta santa casa, si hay alguien que sea verdaderamente importante es el lector. Dicho esto, no extrañará a nadie la política que viene manteniendo últimamente nuestra editorial de lanzar al mercado una serie de números especiales que tratan de manera monográfica aquellos temas que suscitan especial interés entre nuestros lectores.

Continuando, pues, con esta tónica que se inició hace un par de meses con la aparición de nuestro ESPECIAL CODIGO MAQUINA y a la vista de la amplia aceptación que ha tenido este monográfico entre nuestro público, nos hemos decidido a elaborar un nuevo especial, que aparecerá con el logo de nuestra revista hermana MSX CLUB destinado esta vez el software.

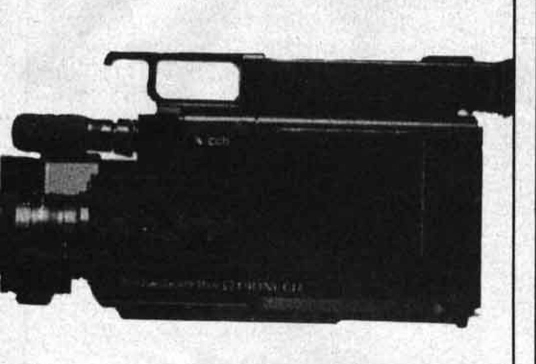
En este ESPECIAL SOFTWARE se analizarán todos aquellos puntos que interesan al usuario, además de elaborarse por vez primera el catálogo general de software para el sistema MSX.

Esperamos así cubrir un vacío que había en este ámbito, y proporcionar de esta manera a nuestros amigos lectores infor-



mación detallada y precisa acerca de los programas que puede hacer correr en su ordenador.

Que no se te olvide. En el momento en que vean la luz pública estas páginas, ya estará en tu quiosco. Cómpralo, está todo aquello que deseas saber.



VE-M800 PIONEER

Cámara filmadora

La cámara filmadora de vídeo Pioneer VE-M800 de 8 mm es uno de los aparatos más interesantes para captar imágenes tanto familiares como de cualquier naturaleza y que también puede transformarse en un original periférico de tu MSX. En efecto, esta cámara que trabaja con el estándar mundial de 8 mm dispone de autofocus y alta calidad de grabación de vídeo y audio, dando excelentes resultados de imagen y sonido. Con ella puedes obtener imágenes, que luego podrás procesar y tratar a través del digitalizador de vídeo del Pioneer PX-7 o de los super imposer de los MSX-2.

CODIGOS POSTALES

En el mes de agosto, ha entrado en vigor la nueva codificación Postal española. Como sabéis, hasta ese momento tan sólo las capitales de provincia tenían asignado un código postal.

En esta segunda fase puesta en marcha este verano por Correos, se ha asignado codificación postal a todas las localidades españolas. Por lo tanto, rogamos encarecidamente a todos aquellos de nuestro lectores que hayan de comunicarse por cualquier asunto con nosotros (suscripciones, números atrasados, libros, cassettes, programas) tengan a bien indicar de una manera clara su código postal, puesto que en caso de no hacerlo no podemos hacernos cargo de la demora o incluso del extravío de nuestros envíos. Por favor, esto es muy importante: Indicad de una manera clara y legible vuestro código postal; en definitiva redundará en beneficio de todos.

Por otra parte, rogamos un poco de paciencia a todos aquellos que nos escribieron en el interín de la aplicación de esta nueva codificación; es posible que nuestros envíos del mes de agosto sufran una pequeña demora por este mismo motivo. A ellos les pedimos disculpas en nombre de Correos.

TRUCOS DEL PROGRAMADOR



VIDAS INFINITAS

Parece que hay un interés general por los POKEs que sirven para trucar juegos y hacerlos más cómodos. Pues bien a través de estas páginas intentaremos complaceros y empezaremos con dos arcades: BLAGGER, de Aligata, y MUTANT MONTY, de Artic Computing.

El primero es, sin duda, muy popular debido a que se regala con la compra de algunos modelos de ordenadores PHILIPS MSX. Consta de 20 pantallas diferentes y llegar a la última resulta difícilmente alcanzable, salvo, claro está, que se disponga de inmortalidad o de algún otro truco.

El siguiente cargador te dará a elegir entre jugar con vidas infinitas o hacerlo con un número finito de ellas. Debes teclearlo y ejecutarlo y dejar que el programa se ocupe de cargar a BLAGGER.

```
10 KEYOFF:CLS
20 PRINT"(QUIERES VIDAS
 INFINITAS?  (S/N)
30 Z$=INPUT$(1):IFZ$="N
"ORZ$="n"THEN70
40-IFZ$="S"ORZ$="s"THEN
50ELSE30
50 D1=&H9C3D:D2=&H9CDD
60 DT=201:GOTO100
70 CLS:INPUT"NUMERO DE
VIDAS";DT
80 IFDT>60THEN70
90 D1=&H9254:D2=D1
100 SCREEN1,0,0,1::BLOA
D"blagr",R:BLOAD"CAS:"
110 DEFUSR=37202!:POKED
1,DT:POKED2,DT:A=USR(0)
```

MUTAN MONTY es otro juego de arcade con un gran número de pantallas (40). Sin embargo, acabarlo por completo es tarea casi divina. Estamos convencidos de que no lo conseguirás con menos de 20 vidas. En fin, el siguiente cargador te dará la oportunidad de preseleccionar el número de vidas o la inmortalidad. Tecléalo y ejecútalo antes de poner la cinta de MUTAN MONTY en el reproductor.

```
10 KEYOFF:CLS
20 PRINT"(QUIERES VIDAS
 INFINITAS?  (S/N)
30 Z$=INPUT$(1):IFZ$="N
"ORZ$="n"THEN70
40 IFZ$="S"ORZ$="s"THEN
50ELSE30
50 D1=&H97BC:D2=D1
60 GOTO100
70 CLS:INPUT"NUMERO DE
VIDAS";DT
80 IFDT>99THEN70
90 D1=&H9321:D2=&H965B
100 BLOAD"CAS:"
110 DEFUSR=36864!:POKED
1,DT:POKED2,DT:A=USR(0)
```

TECLAS DE FUNCIÓN

Aprovecho para darles algunos trucos para su sección de trucos del programador.

El primero es referente a las teclas de función y es el siguiente siempre que se desee que cuando se toque una tecla de función ésta se ejecute prescindiendo de darle a return se hace lo siguiente.

KEY1,"LIST 50–"+ CHR$(13)

Este trozo que se le añade es el que hace que se ejecute. El otro truco es para cuando se desconoce el nombre de un programa haciendo lo siguiente entrará cualquier programa.

Ej.: BLOAD"CAS:",R y entran todos los que estén en código máquina.

LOAD"CAS:",R y CLOAD"" de este modo entrarán los programas sin necvesidad de poner el nombre del programa.

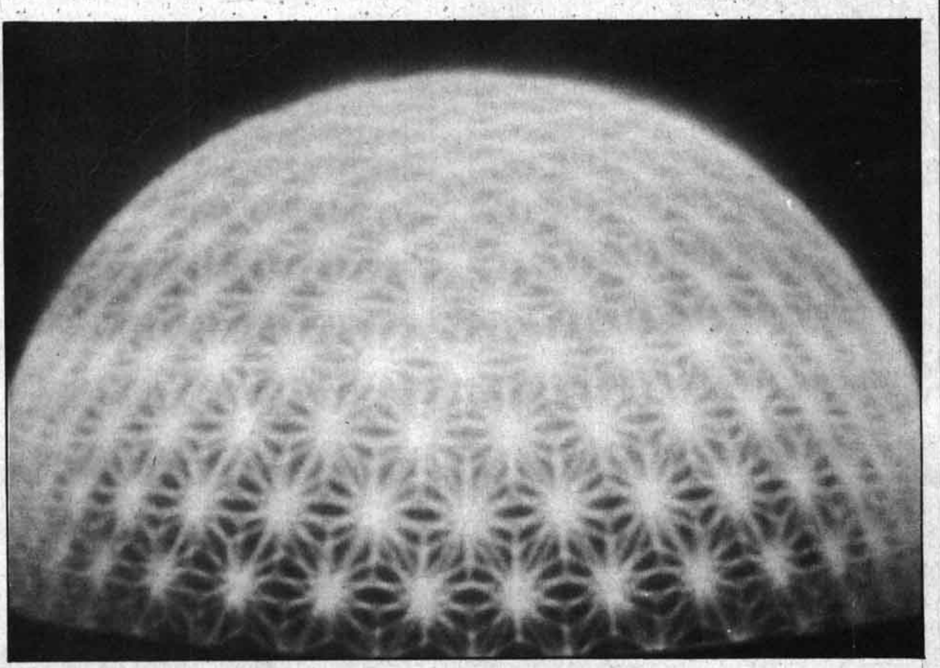
Gonzalo Rovira Bazán

CÓDIGO DECIMAL

No siempre se dispone de la tabla de códigos decimales de los caracteres. Por este motivo os presento este pequeño programa, que os ayudará en el momento de hacer vuestros programas. Tecléalo, haz run, enter, y pulsa cualquier tecla o carácter, el ordenador te indicará su código decimal.

```
1 CLS: KEY OFF: SCREEN 0:
  COLOR 1,4
2 E$= INKEY$: IF E$=""
  THEN 2
3 FOR I=0 TO 255
4 IF E$= CHR$ (I) THEN
  PRINT " El código deci-
  mal de ";CHR$(I);
  : PRINT" es :";I
5 NEXT
6 GOTO 2
```

Sergi Pechola Jansa



PRECIO CANARIAS 275 PTAS.

MSX EXTRA

LA PRIMERA REVISTA DE MSX DE ESPAÑA

NUMERO ESPECIAL - P.V.P. 275 PTAS (incluido IVA)

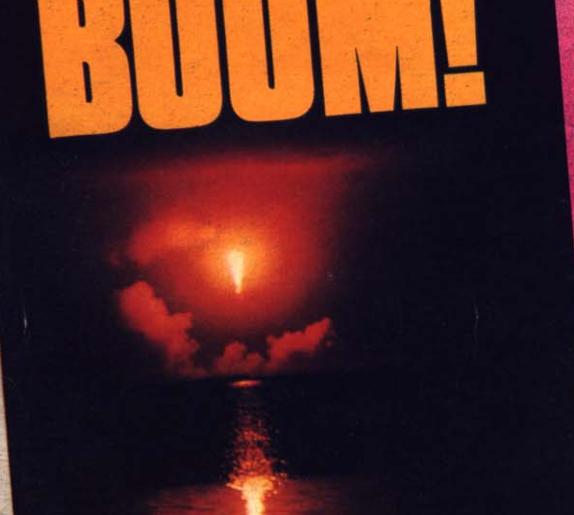
Especial

Código Máquina

OnLine
COMPUTERS PLUS®

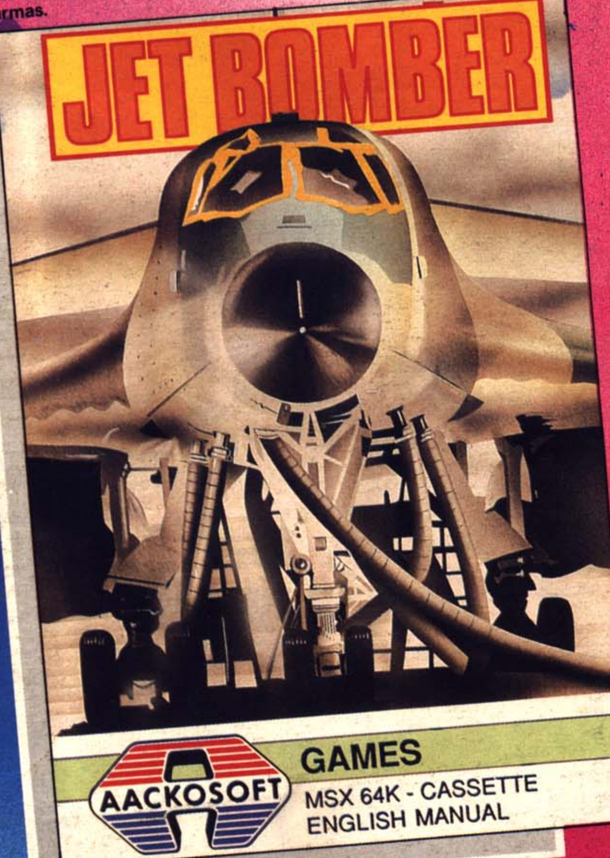
¡¡¡entra en juego!!!

JET BOMBER
Una simulación de acción interactiva con gráficos en 3 dimensiones y voz digitalizada que te empujaran a los limites de tu destreza...
La situación: La revolución llega a su punto de ebullición y la violencia traspasa los desiertos y llega a la República Unida de Abadeer...
La solución: Se necesita un piloto de inigualable destreza y con nervios de acero, para volar secretamente a Abadeer, los últimos soldados y armas.





MSX 16K
GAME CASSETTE





¡¡BOOM!!
Un excitante juego de invasión del espacio. Ve y destrúyelos en el espacio.
Controles: mueve tu palanca para arriba, para abajo, izquierda o derecha. Presiona el botón de juego.

GALERIAS
Marcando estilo.